Workgroup: Industrial Internet of Things
Internet-Draft:
draft-tang-iiot-industrial-scheduling-01
Published: 2 November 2020
Intended Status: Informational
Expires: 6 May 2021
Authors: C. Tang — Chongqing University
S. Ruan — ChongQing University
B. Huang — ChongQing University
H. Wen — ChongQing University
X. Feng — ChongQing University

Research on Multi-Priority Scheduling Technology for Industrial Field and Cloud Interconnection

Abstract

This document describes the multi-priority scheduling technology of industrial field and cloud interconnection under the application of 5G communication, including spectrum resource scheduling based on 5G slice in the access process of industrial data and task collaborative scheduling based on edge computing.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

5G mobile communication develops rapidly driven by different application scenarios and diversified service deployment. Industrial Internet based on 5G technology has also accelerated research and deployment. To give full play to the role of 5G technology in the industrial system, the first prerequisite is to realize the interconnection between the industrial site and the cloud. On the one hand, 5G spectrum access is limited, on the other hand, due to the constraints of industrial equipment computing resources, we typically offload a portion of our industrial applications so that we have enough computing resources on our computing system to execute, such as micro server, cloud server, or data center. Therefore, the multi-priority scheduling technology of industrial field data to the cloud is an important key to be solved.

In the industrial field, industrial field data mainly refers to the real-time data collected by industrial production equipment and target products under the operation mode of the Internet of Things, including the data reflecting the operation state of equipment and products, such as operation and operation condition, working condition and environmental parameters. These data can be uploaded to the cloud for data processing and analysis through 5G base station, and then the data can be reused by users for intelligent design, intelligent production, networked collaborative manufacturing, intelligent service and personalized customization.

In the future smart factories, there will be a huge demand for industrial field applications, including Internet of Things data acquisition, intelligent robots, industrial AR and other services.It is precisely because many applications have different demands for network service quality that it is necessary to study the multi-priority scheduling technology based on service quality for the data of these different demands.

From the industrial site to the cloud, priority scheduling problem can be decomposed into two parts. The first part includes the allocation of spectrum resources corresponding to the schedule for different priority services in the process of industrial data access through 5G communication technology. For this purpose, we propose an uplink scheduling algorithm for industrial field data based on 5G slice. The other part includes the allocation and scheduling of computing resources for tasks in edge computing nodes and cloud computing nodes. For this reason, we propose a collaborative scheduling algorithm for big data tasks in industrial field based on edge computing.

1.1. Application requirements of industrial Internet

1.1.1. Internet of Things data collection

The Internet of Things, is the Internet of Everything. Internet of Things data contains information needed by sound, light, heat, electricity, mechanics, chemistry, biology and location, etc. Its goal is to combine all kinds of information sensing devices with the Internet to realize the interconnection of people, machines and things at any time, any place. Therefore, massive machine communication brings great demand for network coverage. The information collection of industrial control system includes the sensor data designed in the industrial system and mainly collects the physical events and data occurring in industrial production and manufacturing factories, including various physical quantities, identification, positioning and other data.

1.1.2. Intelligent robot

Machine vision has become more and more popular in manufacturing enterprises, such as automobile factories, through machine vision to detect product defects. This kind of application requires a lot of network bandwidth. And since intelligent robots need to complete corresponding intelligent operations, the fast response of highly reliable 5G network is also a prerequisite.

1.1.3. Industrial AR

5G AR will become an important application of industrial Internet. When combined, the two can be applied to multiple scenes in the industrial field, including: man-machine collaboration, monitoring of production process, pre-job training for new employees, product quality detection, remote assistance and guidance, etc. For example, when industrial equipment damage alerts the network for maintenance, remote technicians can remotely guide on-site technicians using head-mounted AR equipment through the industrial AR application to complete the equipment maintenance process. This makes the industrial network need to provide reliable network bandwidth guarantee and delay demand.

1.1.4. Other businesses

In addition, there are other business needs in the industrial field, including security monitoring and other businesses. Different businesses have different requirements for network services.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in ERROR: Undefined target: RFC2119.

3. Uplink scheduling algorithm based on 5G slice

In the access stage from industrial field to 5G NR, due to the service demand and communication quality of different data on site, the base station side needs to carry out multi-priority uplink scheduling for different service data and allocate spectrum resources.

In this paper, inter-slice scheduling and intra-slice user scheduling are used for uplink resource allocation. On the basis of satisfying the service quality required by 5G different slice scenarios and different industrial field data, the fairness and throughput of the scheduling algorithm are improved.

3.1. Common Upline scheduling of 5G NR:

Firstly, when UE needs to send uplink data, it will first put the required data into the cache, and then submit its BSR (Buffer State Report) to the base station through PUCCH (Physical Uplink Control Channel. At the same time, the Scheduling Request is sent to inform the base station gNB that it needs to send data.

Secondly, the uplink scheduler of gNB receives an uplink scheduling request from UE, which allocates resources to UE based on the UE's cache status report and the uplink channel condition of UE, which is obtained by the SRS Reference Signal that UE periodically sends to the gNB. Distribution results are sent to UE by PDCCH (Physical Downlink Control Channel) using UL Grant (Uplink Grant). Thirdly, UE sends data to the base station through PUSCH (Physical Uplink Shared Channel) using the resources allocated by the base station.

The uplink scheduler of gNB receives the cache status report and upline channel status of UE and completes the dynamic scheduling of time-frequency resources according to the built-in scheduling algorithm. There are three common scheduling algorithms: Round-Robin(RR) algorithm, Max C / Ι algorithm, Proportional Fairness(PF) algorithm.

The RR algorithm allocates resources for different users of request scheduling in a circular way. This algorithm only considers the fairness among users and loses the system throughput. The Max C / I algorithm always provides resources for the best users of the channel, which can maximize the system throughput but cannot guarantee fairness between cell users. PF algorithm considers the ratio of instantaneous rate and long-term average rate when selecting users, and adjusts different users by using weight value to achieve the purpose of giving consideration to the overall throughput of the system and fairness of users at the same time, but does not consider QoS information of service.

The explosive growth of data rate and capacity demand, as well as the large-scale, high reliability, low delay, and other differentiated demands have brought about the development of 5G. Therefore, faced with different industrial scenarios, different QoS requirements of the business, A more reasonable multi-priority scheduling algorithm needs to be designed. Under the condition of limited wireless resources, the algorithm makes reasonable allocation of 5G wireless resources in different segments, so as to meet the service requirements of high-priority services of intelligent factories as far as possible and improve the resource utilization rate and fairness among users.

This section provides a multi-priority resource scheduling method for industrial field data to improve resource utilization as much as possible on the basis of meeting the service quality required by different industrial field services under different slice scenarios of 5G.

3.2. Uplink scheduling algorithm flow

The general flow of the uplink scheduling algorithm based on 5G slice is as follows:

Step 1: During a scheduling cycle, determine if the task cache queue is empty. If null, wait for the next scheduling cycle; if non-null, proceed to the next step.

Step 2: Use the inter-slice scheduling algorithm to allocate resources to the three network slices according to requirements.

Step 3: For the resources obtained from each slice, resource scheduling is carried out for each user in the slice. When complete, wait for the next scheduling cycle.

3.3. Inter-slice scheduling

For Step 2, inter-slice resource scheduling needs to meet:

1. The resources obtained from different slices are isolated and independent in the frequency domain, and can be adjusted flexibly. The congestion of one network slice does not affect other network slices.

2. By allocating spectrum resources with better channel conditions to the high-priority slice, the throughput of the system and the service guarantee for high-priority service can be improved.

Assuming that the number of RB (Resource blocks) that the scheduler can configure is , and at the time of scheduling, the total number of users requesting resources is , the priority of each slice in each RB is defined in this paper by formula (1) :

$$P(i,j) = R(i,j) \tag{1}$$

Where,P(i,j) represents the priority of the i-th slice at the j-th RB block in one scheduling cycle. The greater P(i,j) indicates that the scheduling priority of i-th user in j-th RB block is higher. In order to improve system throughput, the priority is based on the rate and calculation of all users in the j-th RB block in the i-th slice. R(i,j) represents the rate sum of all users in the j-th RB block in the i-th slice According to the calculation, we can get the priority matrix.

For the service requirements of different businesses of industrial field data, the uRLLC slice needs low delay and high reliability, such as a remote real-time cooperative robot, which needs to give priority to allocation of resources and reduce queuing delay. The eMMB slice will have a lot of high data volume business requirements, priority allocation of resources will improve the overall network throughput, but doesn't have such as high requirements for the delay as the uRLLC slice. The mMTC slice has the lowest scheduling priority, and in most cases, the amount of uplink data is not large, and the delay requirement is low. Therefore, according to the order of the uRLLC slice, the eMMB slice, and the mMTC slice, RB resources are configured according to the priority matrix.

According to the resource scheduling requirements of slices, the final inter-slice resource scheduling scheme is as follows:

Step 1: Calculate the priority matrix according to formula (1);

Step 2: According to the priority order of slices, select the i-th slice for resource scheduling, and i is initialized to 1;

Step 3: Select the i-th slice, the j-th RB block with priority ranking, and j is initialized to 1;

Step 4: Determine whether the currently scheduled RB block is adjacent to the RB that the slice has been allocated. If yes, perform step 5. If no, set j=j+1 and repeat step 3.

Step 5: Assign priority j-th RB to the i-th slice and remove that RB from the RB queue;

Step 6: Determine whether the resource request for the i-th slice has obtained enough resources. If so, perform step 7. If not, set j=j+1 and re-perform step 3.

Step 7: Determine whether the RB sequence is empty or whether all slices have obtained sufficient resources; if so, end slice resource scheduling; if not, execute i = i +1 and repeat step 2.

3.4. Intra-slice user scheduling

After resource scheduling among slices, these slices obtaine their respective continuous and isolated RB groups, and then the intra-slice user scheduling will not interfere with each other.

The user scheduling in the slice can be understood as a logical cell, and the scheduler will conduct resource scheduling on the users belonging to this cell through the resources obtained by intra-slice scheduling. Because of different QoS requirements of data in 5G industrial field, performance indicators that need to be comprehensively considered in the process of user scheduling in the slice include transmission rate, delay demand, packet loss rate and the amount of data to be transmitted.

The priority of i-th user in the j-th RB group at time t is calculated by the formula (2) :

$$P(i, j) = -\log(p(i))/Td(i) * r(i, j)/R(i) * d(i)/D \quad (2)$$

p(i) represents the maximum rate of packet loss of user i, p(i) in (0,1). Therefore, -log(p(i)) indicates the lower the maximum loss rate, the higher the priority. Td(i) represents the maximum wait delay for the user i, The smaller Td(i)the higher the user priority. r(i, j)represents the instantaneous transfer rate of the i-th user in the j-th RB group during the t scheduling cycle. R(i)represents the average transmission rate before the i-th user. The higher the instantaneous transmission rate is, the better the channel quality condition is, and the higher the priority is. d(i)represents the amount of data that user i is waiting to send at time t. The higher d(i)/D is, the higher the proportion of the pending business volume of user i in the total business volume of all requesting users at time t is, and the higher the priority is.

Therefore, the intra-slice user scheduling scheme is as follows:

Step 1: Complete the inter-slice scheduling.

Step 2: For all RB of a single slice, it is divided into the same size RB group according to the number of slice users.

Step 3: Calculate the priority of each user in the slice on each RB group according to Formula 2.

Step 4: Assign the RB group with the highest priority to each user in turn according to the user priority.

Step 5: Determine whether the RB sequence is empty, if so, the resource allocation ends, if not, repeat step 4.

3.5. The summary of this chapter

The algorithm proposed in this section has the following advantages:

1. Analyze the different service requirements of different services for industrial field data in 5G environment. The inter-slice scheduling algorithm is used to complete the resource allocation of 5G three slices to ensure the flexible scheduling and isolation of resources between slices. Ensure that the required resources for high priority businesses such as the uRLLC slice business are allocated and improve the throughput of the system.

2. The intra-slice scheduling comprehensively considers the data service transmission rate, delay demand, packet loss rate, data volume to be transmitted, and other performance indicators, and gives priority to the scheduling of users with good channel conditions, high delay requirements, high-reliability requirements, and large data volume to be sent.

4. Collaborative scheduling algorithm based on edge computing for big data task in industrial field

With the rapid development of industrial Internet and mobile communication technology, applications such as face recognition, short video traffic, autonomous driving, drone operations, industrial detection, and other applications having higher requirements for computing, Relying only on the current centralized cloud computing architecture model, The computing power to provide business is not enough. Facing the continuous generation of big data in industrial production, entertainment, education, and other industries, there is an urgent need for cloud-centric computing architecture to expand to a distributed computing service architecture, and the most representative of these are edge computing and fog computing model.

With the emergence of big data, the computing power of mobile terminals has also begun to rise. According to the type of data and service quality requirements, higher requirements are put forward for computing speed and processing capacity. Some tasks with particularly high latency requirements are suitable for distributed processing mechanisms, relying on cloud processing, and real-time performance cannot be met under conditions of heavy network load. Therefore, it is necessary to rely on edge computing to sink computing power, and dynamically allocate computing resources based on tasks and real-time performance. In the industrial field environment, there are various sensor data, and the corresponding instruction requirements generated by them, and service processing. To a certain extent, the computing power of field devices is insufficient, and the amount of data presents a complex and huge trend. It is difficult to only rely on edge terminal equipment to implement business logic. At the same time, in complex industrial sites, the response requirements of various services are inconsistent, so the system's response capabilities, processing capabilities, and throughput capabilities have a huge test.

Therefore, in the industrial Internet big data scenario, the collaboration of tasks, the allocation of resources, and the efficient processing of data have a huge research space and also have important practical value, attracting the attention of many scholars. However, the traditional research point is to think about the resource allocation and utilization of edge nodes, as well as the coordinated scheduling of the edge and the cloud, or to focus on the issue of task priority. However, these algorithms have drawbacks, either considering the resource allocation problem between nodes, or only considering the task priority problem, and not coordinated consideration, at the same time, the network link bandwidth also has an impact on the system. Therefore, it is necessary to propose a better task scheduling algorithm based on task requirements, combined with edge computing, network bandwidth, and task real-time requirements, to maximize resource utilization and user satisfaction.

4.1. Task collaborative scheduling algorithm flow

Based on the complex industrial site environment, task requirements and the different characteristics of the amount of calculation, from a new perspective, comprehensive consideration of computing resources, user satisfaction, through edge computing technology to achieve business tasks between the terminal and edge server resource scheduling problem, in In the case of meeting the minimum resources, user satisfaction can also be guaranteed to meet the real-time task processing generated in a variety of industrial production processes.

Based on the actual needs of industrial field task data, we considered and provided a collaborative scheduling algorithm for industrial field big data tasks based on edge computing. As shown in Figure 4, the method steps are: the terminal publishes the service, and the scheduler obtains the service information. Such as calculating the number of tasks, delay requirements. According to the task delay requirements, the task's urgency and priority are evaluated. According to the current bandwidth resources, it is determined whether the task should be offloaded to the edge server. The task cache status of the scheduler at this time and the computing information of each edge server are obtained. The number of queued processing is based on system resource status, current business task volume, delay requirements, etc. to perform reasonable task scheduling of the server and terminal, and repeat the execution until all tasks are allocated and executed accordingly.

Specific steps are as follows:

Step 1: The terminal publishes the service, and the scheduler obtains the service information, such as the number of calculation tasks , the delay requirements:T(i),with i=1 ... n.

Step 2: The task priority: LEVEL(i), with i=1 ... n,. is evaluated according to the time delay requirements of the task, which has an impact on the subsequent task scheduling and further reflects user satisfaction.

Step 3: Obtain the current network link information, that is, the remaining bandwidth of the network: R. Assuming that the upload rate of the task is α , the task upload needs to meet, if the current bandwidth is not enough, the task needs to be unloaded and the waiting delay is recorded.

Step 4: Obtain the task scheduling threshold of the scheduler at this time, as well as the calculation information of each edge server, the number of tasks queued, and the queue waiting for the delay.

Step 5: Perform reasonable task scheduling of the server and terminal according to the state of system resources, current business task volume, and delay requirements.

4.2. Scheduling strategy process

The following is the processing flow of the scheduling strategy: according to the machine learning algorithm, the task delay requirements are comprehensively considered, the remaining capacity of the terminal, the remaining computing capacity of the edge server, and the total delay of the task assigned to the terminal. The task is assigned to the edge server to calculate the total delay, the output layer is maximized through the fully connected layer, and finally, the softmax layer estimates the probability of assigning to the terminal or the edge server. Therefore, the internal parameters of the network are learnable parameters, so it can be based on system resources conditions, network load and adaptive adjustment parameters provide a basis for subsequent optimization. The number of iterations can be determined by yourself. Finally, the updated parameters are used to allocate real resources to the tasks in the current scheduler.

4.3. The summary of this chapter

Compared with the existing industrial field task resource collaborative scheduling algorithm, the algorithm proposed in this section has the following advantages:

Realize the reasonable scheduling of industrial field tasks in terminal equipment and edge servers, fully consider the system's computing resources, improve the system's ability to process tasks, and minimize Reduce the resource consumption of the system and avoid the calculation waiting for behavior caused by the unbalanced resource allocation; consider the execution status of the system, the task calculation amount, and the delay requirements for optimal scheduling when performing task scheduling; comprehensively consider the construction of new optimization goals, In order to achieve system resource utilization efficiency and user satisfaction.

5. Security Considerations

5.1. Security Considerations

For edge computing equipment, security problems caused by indirect or self-inflicted causes during operation (e.g. energy supply;Cooling and dust removal, equipment loss, etc.), although the operation threat is not as complete as the damage caused by natural disasters, the lack of a good response means will still lead to disastrous consequences, resulting in the performance degradation of edge computing, service interruption and data loss.Especially in the industrial Internet scene, the factory is more professional in the maintenance and overhaul of its own equipment, but IT is difficult to timely deal with the operation and maintenance of IT equipment.

5.2. Network security requirements

Compared with cloud computing data centers, edge nodes have limited capabilities and are more vulnerable to hackers. Although the damage of a single damaged edge node is not great, and the network has the ability to quickly find nearby alternative nodes; But if hackers use the compromised edge nodes as "broilers" to attack other servers, it could affect the entire network. Most of the existing security protection technologies have complex computational protection processes, which are not suitable for edge computing scenarios. Therefore, it is a great demand for network security to design lightweight security technology suitable for edge computing architecture in industrial Internet scene.

5.3. Data security requirements

In edge computing, users outsource data to edge nodes and transfer control of data to edge nodes, which introduces the same security threats as cloud computing. First, it is difficult to ensure the confidentiality and integrity of the data because the outsourced data may be lost or modified incorrectly. Second, unauthorized parties may misuse the uploaded data to seek other benefits. Compared with the cloud, edge computing has avoided the long-distance transmission of multiple routes and greatly reduced the outsourcing risk. Therefore, the security problem of data belonging to edge computing is increasingly prominent. For example, in such a complex and changeable environment, how to realize the safe and rapid migration of data after the collapse of an edge node.

5.4. Security Considerations

Application security, as the name implies, guarantees the security of the application process and results. In the era of marginal big data processing, by moving more and more application services from cloud computing centers to network edge nodes, applications can be guaranteed to get shorter response time and higher reliability, and meanwhile, network transmission bandwidth and intelligent terminal power consumption can be greatly saved. However, edge computing not only has common application security problems in information systems, such as denial of service attack, unauthorized access, software vulnerability, abuse of authority, identity impersonation, etc., but also has other application security requirements due to its own characteristics. In the scenario where multiple security domains and access networks coexist at the edge, how to manage user identity and realize authorized access to resources become very important to ensure application security.

6. Acknowledgements

This template was derived from an initial version written by Pekka Savola and contributed by him to the xml2rfc project.

This document is part of a plan to make xml2rfc indispensable [DOMINATION] .

7. IANA Considerations

This memo includes no request to IANA.

All drafts are required to have an IANA considerations section (see Guidelines for Writing an IANA Considerations Section in RFCs [RFC5226] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

8. References

8.1. Normative References

[min_ref] authSurName, authInitials., "Minimal Reference", 2006.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

8.2. Informative References

[DOMINATION] Mad Dominators, Inc., "Ultimate Plan for Taking Over the World", 1984, <http://www.example.com/dominator.html>.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, DOI 10.17487/RFC2629, June 1999, <https://www.rfc-editor.org/info/rfc2629>.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <https://www.rfc-editor.org/info/rfc5226>.

Appendix A. Additional Stuff

This becomes an Appendix.

Authors' Addresses

Chaowei Tang
ChongQing University
No.174 Shazheng Street, Shapingba District
Chongqing
400044
China

Email: cwtang@cqu.edu.cn

Ruan Shuai
ChongQing University
No.174 Shazheng Street, Shapingba District
ChongQing

China

Phone: +86 189-6826-0296
Email: rs@cqu.edu.cn

Huang Baojin
ChongQing University
No.174 Shazheng Street, Shapingba District
ChongQing
China

Email: baojing-huang@foxmail.com

Wen Haotian
ChongQing University
No.174 Shazheng Street, Shapingba District
ChongQing
China

Email: wenhaotianrye@foxmail.com

Feng Xinxin
ChongQing University
No.174 Shazheng Street, Shapingba District
ChongQing
China

Email: xxfeng@cqu.edu.cn