Scholar Publishing Group
Distributed Processing System
https://doi.org/10.38007/DPS.2022.030404
ISSN 2790-0916 Vol. 3, Issue 4: 28-35

Fault Tolerant Task Scheduling Algorithm for Distributed System Based on EDF

Hedjazi Djalal*

CFIN, Norrebrogade 44,Build 10G,4th, DK-8000 Aarhus, Denmark

**corresponding author*

Keywords: EDF Technology, Distributed System, Fault-tolerant Task, Scheduling Algorithm

Abstract: With the continuous development of social economy, enterprises have higher and higher requirements for production efficiency and product quality. How to complete the task assignment under the basic constraints of ensuring production capacity and low cost is the main problem faced by the manufacturing manufacturing industry. This paper presents a fault-tolerant strategy for distributed systems. Firstly, the paper gives the research background and significance. Secondly, it introduces the layered scheduling model based on EDF and the distributed system with weight weighting designed by particle swarm optimization algorithm. The simulation scheme is tested and analyzed by MATLAB software to verify the results. Finally, it is concluded that the scheme can fully ensure the stability of the scheduling process in the actual working environment. The stall error rate is also guaranteed to be within the range of failure, so it can be widely used in various fields.

1. Introduction

With the development of computer technology, fault-tolerant scheduling using distributed systems has become a very hot topic in the research field. Due to the continuous expansion of the enterprise's production scale, the acceleration of product upgrading and the improvement of customers' requirements for service quality, the traditional task allocation method can no longer meet the complex and changeable demands in the market competition environment [1-2]. In the multi-objective decision-making optimization problem, the most common is to analyze the fault-tolerant scheduling algorithm for the selection process of the combined resource allocation scheme, which has become one of the most important and effective methods with the best universality [3-4].

With the rapid development of distributed system, the research on it has become a hot issue. American scholars have given a definition of equipment selection strategy in a distributed system.

When the available resources in a region are analyzed and calculated and then allocated to a specific task (or activity) and corresponding resources need to be configured, priority will be given to the task or activity that requires the lowest cost and can achieve the maximum optimized operation efficiency, that is, the optimal scheduling scheme [4-5]. The domestic research on distributed system started late, and it was not until the 1990s that we began to explore it deeply. At present, the application of distributed systems in China is mainly concentrated in industrial enterprises and commercial fields. Some scholars proposed a heuristic particle swarm optimization (PID) based allocation method for traditional probabilistic scheduling problems, and proved through simulation that this strategy can improve work efficiency, reduce production costs and reduce resource waste to achieve the goals, and also provide ideas and technical support for future large-scale deployment of fault-tolerant tasks [6-7]. Therefore, based on EDF, this paper designs a fault-tolerant task scheduling algorithm for distributed systems.

With the continuous progress of society and the rapid development of network technology, the competition between enterprises becomes more and more fierce. For a company which is developing rapidly and has a certain scale and high profit, how to realize resource sharing is an urgent problem to be solved. In order to solve this problem, distributed system is proposed in this paper as an important method to deal with some large-scale complex tasks (such as meetings, scheduling, etc.), which may occur in production but can not meet the needs well or can not allocate effective work time under certain conditions. On this basis, heuristic search algorithm is used to optimize the solution.

2. Discussion on Fault Tolerant Task Scheduling Algorithm for Distributed System Based on EDF

2.1. Fault Tolerance of Distributed System

The implementation fault tolerance problem of distributed system refers to how to allocate the proportion of each task in the whole life cycle when a fault occurs at a certain time, so that the whole system is in a safe and stable operation state [8-9]. Scheduling problem is a typical and complex assignment model, which involves multiple tasks. What needs to be considered in this process is that when the scheduling fails, the entire fault-tolerant task will fail. (1) Fault tolerance. This strategy aims at the overall optimization of the system and allocates resources for each execution node. (2) The characteristics of high load characteristics and dynamic performance requirements, fast response speed and low cost will affect the quality of the system, so it is necessary to ensure that it can effectively solve these problems. For the two common events mentioned above, one is scheduling failure and the other is failure to start. What we are talking about here is a random event (such as equipment failure and other emergencies when there is no equipment configuration on different time nodes) or an accident caused by a failure when a task cannot be started after the scheduling is successful. During the implementation of the distributed system, each area needs to set a certain number of tasks to allocate corresponding areas to complete specific functions, but these tasks are usually ignored in the schedule. If we consider the actual problems, the distributed system may not operate normally. Fig. 1 is a general model of scheduling problem.

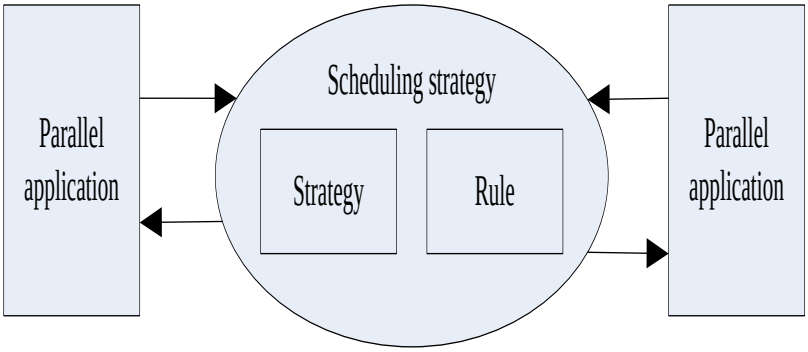

Figure 1. A general model of the scheduling problems

2.2. Fault Tolerance Strategy for Distributed System

There are mainly two fault-tolerant strategies for distributed systems. One is to use existing resources, that is, to design a reasonable, economical and efficient scheme by analyzing existing resources. There are many feasible options in these alternative missing sets and their sizes are constrained. For example, some uncertain problems may be encountered in actual situations, or it is impossible to know whether there are alternative solutions, or there is no clear boundary value. The other is to optimize the existing feasible solution and generate a new method. It is user-centered and decides whether to use this scheme or other feasible solutions as the final goal according to the scheduling object and scheduling problem itself. For large enterprises, the first consideration is their scale, production capacity and customer distribution, and the second is what kind of methods to deal with different types of faults. When fault-tolerant tasks occur, the dispatcher in the system will allocate resources according to the user requirements and business requirements in the current scenario. However, since the distributed system is a dynamic random process, the corresponding strategy cannot be given [10-11]. This paper mainly focuses on the research based on EDF algorithm. Firstly, the whole life cycle is divided into three stages: planning stage, implementation stage (preparation stage) and stable operation stage. Then, a fault-tolerant task scheduling model is established and a feasible scheme design flow chart is proposed.

2.3. EDF Algorithm

EDF algorithm mainly uses the existing resources to organize and control the entire network, so it needs to consider its dynamic problems and scheduling constraints to determine the optimal solution or global search strategy, and ensure that the state of each local node is stable and orderly. The algorithm arranges tasks by dividing them into several subgroups, and each group may have one or several points, so that different types of users can obtain the same or similar orders of magnitude of available resources at the same time. However, because each group of scheduling needs a certain degree of time consumption and load intensity (such as working hours, equipment capacity and other factors) to ensure that the system can safely and smoothly achieve packet allocation during operation. It can help us describe the relationship between these rules and phenomena and provide some information for the next processing scheme to support the decision-makers to make correct choices or provide basis for formulating strategies [12-13]. Therefore, the scheduling problem is to solve the problem caused by the uncertainty of random variables, the change of the values of each element in the random range and the different degree of

independence of each order. Before scheduling, the number of nodes and tasks corresponding to each task should be evaluated and analyzed. Then, according to the number of different sub clusters or event generators in the group, determine which data (e.g. output power, etc.) each sub cluster needs to process and calculate all the final possible values as the basis for decision-making. Finally, select those that are most likely to have problems and are not suitable for allocation to each objective function for scheduling through sorting, so as to evaluate and judge the effect required by the whole system [14-15].

$$\Delta T_{\mathrm{e}} = \left(\left(T_e(0) - T_e(m)\right) / T_e(0)\right) * 100\% \tag{1}$$

The improvement of the fault tolerance capability of the fault tolerance priority reduction strategy compared with the fault tolerance capability of the fault tolerance priority inheritance strategy can be measured by Formula 1.

$$\Delta DT_{\mathrm{em}} = \left(\left(T_e(0) - DT_e(m)\right) / T_e(0)\right) * 100\% \tag{2}$$

The fault tolerance of EDF algorithm can be measured by formula 2 compared with the fault tolerance of single fault hybrid strategy.

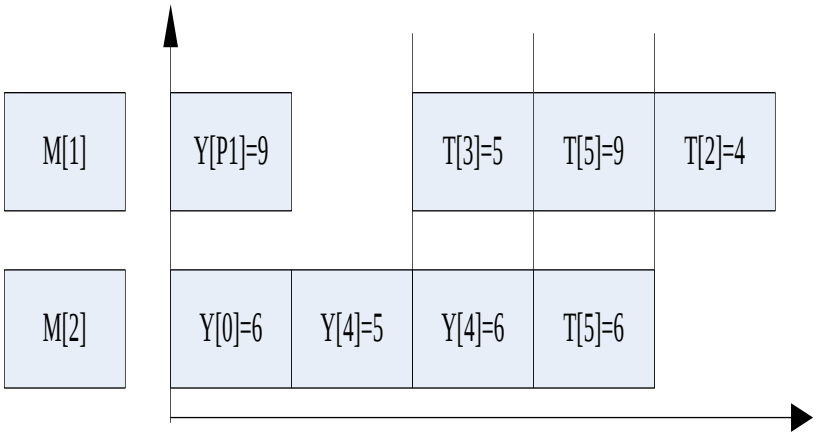

Figure 2. Schedule the algorithm model

Distributed system is based on large-scale parallel computing, which optimizes the whole process in a given environment. The scheduling algorithm model is shown in Figure 2. If anything changes or cannot be adjusted and handled in time, the operation will not continue. This kind of situation is called scheduling problem (NP) or random event, and conversely, it is called emergency situation or unrecoverable state and other event type problems. In a given environment, the whole process is optimized and allocated [16-17].

3. The Experimental Process of Fault Tolerant Task Scheduling Algorithm for Distributed System Based on EDF

3.1. Fault Tolerant Task Model for Distributed System Implementation

On the basis of the above analysis, we propose a fault-tolerant scheduling algorithm for distributed systems. This program enables the user to successfully complete the whole process by

inputting only one task parameter (such as initial value, state vector, etc.) (as shown in Fig. 3). For enterprises of different types and sizes or small companies, there may be multiple files with the same or similar formats. Therefore, in order to improve the system operation efficiency and data storage capacity to meet the business needs, it is appropriate to adopt a multi-level task allocation scheme to store the contents contained in these files. At the same time, it can also be added to the subtasks as an independent unit. When a target function is the maximum value, it is required that all adjacent units must meet this constraint condition at the same time to achieve the desired output results, However, in the actual situation, it is impossible for a node to have a solution set of the point in multiple states, and the optimal allocation scheme obtained by optimizing these data is also different and limited and affected to different degrees.

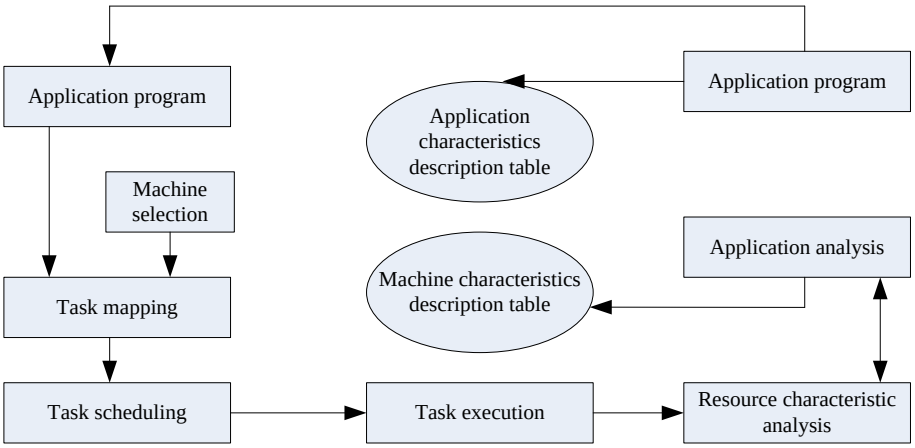

Figure 3. Distributed system implements a fault-tolerant task model

3.2. Performance Test of Fault-tolerant Task Scheduling Algorithm in Distributed System

In order to test the performance of fault-tolerant task scheduling algorithm, we need to simulate the following three aspects. In each period, the actual implementation of each period shall be calculated separately. If a stall error occurs (i.e., the zero offset probability value is less than 0.1), a false trigger failure event will occur. Otherwise, the restart program cannot be started and returned to the initial state or continue to work to the original position to avoid erroneous upgrading of system failure or stopping system update.

(1) Through system analysis, the feasibility of implementing fault-tolerant task scheduling method in distributed system is determined, and the performance test is carried out according to relevant literature and simulation results.

(2) After the fault-tolerant allocation algorithm is designed, the data information such as the program running time, the real-time situation and the actual output results need to be processed.

(3) The redundancy measures generated by different schemes are calculated, and the expected distance method is used to determine whether each optimization item can reach the expected target value. The white box simulated annealing strategy is used to analyze the possible conditions during the implementation of the system, and solutions and suggestions are given to improve the overall performance index.

4. Experimental Analysis of Fault Tolerant Task Scheduling Algorithm for Distributed System Based on EDF

4.1. Performance Test and Analysis of Fault Tolerant Task Scheduling Algorithm in Distributed System

Table 1 is the performance test data of fault-tolerant task scheduling based on EDF algorithm.

Table 1. Fault-tolerant task scheduling

Test times	Availability ratio(%)	Program run time(s)	The shutdown error
1	84%	3	3
2	83%	3	4
3	88%	2	4
4	85%	3	3
5	86%	2	4

Figure 4. Schedule algorithm performance

On the MATLAB platform, the simulation fault-tolerant task scheduling algorithm is established, and the results are compared and analyzed (as shown in Fig. 4). After summing up and analyzing the problems found in the experiment, we can draw the following conclusions: there are some shortcomings in the traditional allocation method that need to be improved, such as large resource constraints, inability to consider the distributed system status and data collection capacity constraints, which will lead to low allocation efficiency and even failure. However, the scheme can fully ensure the stability of the scheduling process in the actual working environment, and the running time is relatively short. The stall error rate is also guaranteed to be within the range of failure, so it can be widely used in various fields.

5. Conclusion

With the popularization of the Internet and the continuous improvement of the technology level of the information industry, distributed systems have been more and more widely used in enterprises. This paper mainly aims at improving the traditional fault-tolerant scheduling algorithm. By analyzing the existing literature and simulation results, it can be seen that for large-scale distributed systems or data models that have failed, a subtask based allocation scheme can be used to solve the problems that the expected objectives (i.e., time constraints) cannot be achieved or the efficiency is low in the actual situation, while for small batch dynamic decision-making processes, time factors need to be considered to ensure that fault-tolerant scheduling is completed within the optimal solution range.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Abhinav Ranjeet Mishra, Rohan N. Kakade, Priyanka Desai Kakade:Dual-Stage EDFA for Improving the Performance of Long-Haul Optical Systems. IEEE Access 10: 13496-13514 (2022).

[2] Lukasz Kruk:Heavy traffic analysis for single-server SRPT and LRPT queues via EDF diffusion limits. Ann. Oper. Res. 310(2): 411-429 (2022).

[3] Seongtae Lee, Sanghyeok Park, Jinkyu Lee:Improved Low Time-Complexity Schedulability Test for Nonpreemptive EDF on a Multiprocessor. IEEE Embed. Syst. Lett. 14(2): 87-90 (2022).

[4] Youngeun Cho, Do Hyung Kim, Daechul Park, Seung Su Lee, Chang-Gun Lee:Optimal Parallelization of Single/Multi-Segment Real-Time Tasks for Global EDF. IEEE Trans. Computers 71(5): 1077-1091 (2022).

[5] Mojtaba Mahdavi, Stefan Weithoffer, Matthias Herrmann, Liang Liu, Ove Edfors, Norbert Wehn, Michael Lentmaier:Spatially Coupled Serially Concatenated Codes: Performance Evaluation and VLSI Design Tradeoffs. IEEE Trans. Circuits Syst. I Regul. Pap. 69(5): 1962-1975 (2022).

[6] Jesús Rodríguez Sánchez, Fredrik Rusek, Ove Edfors, Liang Liu:Distributed and Scalable Uplink Processing for LIS: Algorithm, Architecture, and Design Trade-Offs. IEEE Trans. Signal Process. 70: 2639-2653 (2022).

[7] MinKeun Chung, Liang Liu, Ove Edfors:Phase-Noise Compensation for OFDM Systems Exploiting Coherence Bandwidth: Modeling, Algorithms, and Analysis. IEEE Trans. Wirel. Commun. 21(5): 3040-3056 (2022).

[8] Federico Aromolo, Alessandro Biondi, Geoffrey Nelissen:Response-Time Analysis for Self-Suspending Tasks Under EDF Scheduling (Artifact). Dagstuhl Artifacts Ser. 8(1): 05:1-05:2 (2022).

[9] Husam Abduldaem Mohammed, Muhammad Hafiz Abu Bakar, Siti Barirah Ahmad Anas, Mohd Adzir Mahdi, Mohd. Hanif Yaacob:Real Time in Situ Remote Monitoring for Cladding Modified SMF Integrating Nanocomposite Based Ammonia Sensors Deploying EDFA. IEEE Access 9: 145282-145287 (2021).

[10] Pramit Biswas, Aneek Adhya:Energy-efficient, EDFA lifetime-aware network planning along with virtualized elastic regenerator placement for IP-over-EON. Photonic Netw. Commun. 41(2): 119-135 (2021).

[11] Mitra Mahdiani, Alejandro Masrur:A novel view on bounding execution demand under mixed-criticality EDF. Real Time Syst. 57(1-2): 55-94 (2021).

[12] Daniel Casini, Alessandro Biondi, Giorgio Carlo Buttazzo:Task Splitting and Load Balancing of Dynamic Real-Time Workloads for Semi-Partitioned EDF. IEEE Trans. Computers 70(12): 2168-2181 (2021).

[13] Muris Sarajlic, Ashkan Sheikhi, Liang Liu, Henrik Sjöland, Ove Edfors:Power Scaling Laws for Radio Receiver Front Ends. IEEE Trans. Circuits Syst. I Regul. Pap. 68(5): 2183-2195 (2021).

[14] Rashmi Sharma, Nitin, Mohammed Abdul Rahman AlShehri, Deepak Dahiya:Priority-based joint EDF-RM scheduling algorithm for individual real-time task on distributed systems. J. Supercomput. 77(1): 890-908 (2021).

[15] Juan Vidal Alegría, Fredrik Rusek, Ove Edfors:Trade-Offs in Decentralized Multi-Antenna Architectures: The WAX Decomposition. IEEE Trans. Signal Process. 69: 3627-3641 (2021).

[16] Chakresh Kumar, Ghanendra Kumar:Performance Analysis of Different Modulation Techniques for Super Dense System with RAMAN-EDFA-RAMAN HOA. Wirel. Pers. Commun. 118(1): 343-358 (2021).

[17] Suresh Kumar, Payal:Enhancing Performance of Coherent Optical OFDM FSO Communication Link Using Cascaded EDFA. Wirel. Pers. Commun. 120(2): 1109-1123 (2021).

[18] A. S. Kang, Sanjeev Bhatia, Navneet Kaur, Kuldeepak Singh:Performance evaluation of various hybrid modulation techniques transmitted by EDFA in radio-over-fibre communication. J. Inf. Telecommun. 4(3): 295-313 (2020).