International Journal of Recent Engineering Research and Development (IJRERD)
ISSN: 2455-8761
www.ijrerd.com || Volume 05 – Issue 12 || December 2020 || PP. 13-17

A review on mixed criticality systems research

Yong-xianJin

College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua Zhejiang 321004, China

Abstract: Mixed critical system integrates high security standard and non security critical tasks in the same platform to meet the development requirements of current real-time system hardware platform and software function, and has become one of the important topics in the field of embedded real-time system research. The traditional real-time system does not distinguish tasks of different importance levels, and the execution mode remains unchanged during the running period. Its scheduling strategy only needs to ensure that all tasks are completed within the deadline, which can not adapt to the changes of system execution mode and task time attribute in time. The implementation of mixed critical level system is more complex, and the interaction of tasks with different importance levels in the execution process should be considered, which brings new challenges to the research of task scheduling. At present, the research of mixed critical system has the following problems: when the execution mode of the system changes from low to high, the scheduling strategy usually only ensures the correct execution of the tasks with high importance level, directly discards the low critical level tasks, ignores the conditions for these tasks to continue to be executed, resulting in the destruction of data integrity and the waste of resources; in addition, the current critical level of the system is rarely involved In the research on the change from high to low, it is usually necessary to wait for the processor to be idle before the security degradation is carried out. The system stays in the high critical mode for a long time, which leads to the low schedulability ratio of low critical tasks. In this paper, the research status, existing problems and future research directions of mixed criticality systems are described.

Keywords: mixed criticality system; schedulability; research status

1. The Concept, Background and Significance of Mixed Criticality System

According to the characteristics of the real-time system, the function and time must be met to ensure the normal operation of the system, that is, the task execution must meet the deadline. Real-time systems used in many industrial applications, such as automotive electronics, avionics.Nowadays, with the continuous development of these application fields, the task functions in the system are also gradually diversified, while the corresponding physical platform is developing towards the direction of lightness, and at the same time, safety and energy saving factors should be considered.As a result of these changes, designers need to face more challenges, which require the system to develop towards integration. Integration makes the system characteristics more complex, and requires functional units of different importance levels to share a physical platform. The interference caused by the execution of these functional units makes the operation process of the system more complicated.

This kind of system with multi-level functional units is called mixed critical system. The emergence of new system brings some new problems to task scheduling theory. The traditional real-time scheduling algorithm is only aimed at the real-time system of software and hardware task coordination, and does not distinguish the importance of tasks. If the priority of tasks is assigned only according to the time attribute, the key level inversion may occur [1]. Therefore, the traditional scheduling theory of real-time system is no longer applicable to new systems, and it is urgent to study the scheduling strategy suitable for mixed critical systems.

In 2009, barhost and others wrote the white paper on mixed critical system [2], which formally regarded mixed critical system as an independent research branch, and defined mixed critical system: mixed critical system is a whole including hardware, operating system and middleware, running security critical, mission critical and non-critical application software on the same platform at the same time.

In traditional real-time system, the mission critical level is not distinguished, but the different functions of mixed critical system have different importance. The importance is expressed by the critical level. In practical application, it is necessary to ensure the correct execution of tasks with higher important level. For example, in the automotive electronic system, if the braking function exceeds the deadline, it may cause serious accidents, while some other functions, such as some on-board entertainment functions, will not have a serious impact on the user's safety if the running time exceeds the deadline in an emergency. In order to ensure the correct implementation of the system, it is necessary to authenticate the application of different security level functions in the system. For the functions with high critical level, higher implementation requirements are required, and strict safety critical level certification is required, while the functions with lower importance only need to be certified at lower safety critical level. Certification authorities (CA) in the world divide different functional modules into different key levels. There are many ways to classify the key level: for example, the function of the system is divided into three levels: non critical, mission critical and safety critical in the field of national defense.DO-178b avionics software standard [3] divides tasks into five security assurance levels A to E. IS026262 [4] (International Standard for functional safety of automotive electronic and electrical systems) divides tasks into four safety levels. In addition, IEC 61508, IEC 601-l-4, and D0254 clearly define the classification of different functions in the system.

The standard of mixed critical system is to meet the requirement that all tasks in the system can be successfully scheduled, that is, to meet the deadline. In order to ensure the security of the system, it is necessary to estimate whether the system can schedule all tasks correctly before execution. The worst-case execution time (WCET) method is adopted to analyze the schedulability of tasks based on the WCET. In order to ensure that the system does not have serious security problems, the WCET value corresponding to tasks with higher critical level is also larger, and this part of function is not affected at the critical level promotion stage of the system. The actual execution time of tasks is usually less than that of WCET, and the processor resources can not be fully utilized. Therefore, this verification method to estimate the schedulability ratio of tasks under pessimistic conditions can not effectively schedule tasks. Generally, low critical level tasks are discarded directly after the system critical level is promoted. This processing method is too negative. In fact, the conditions for these tasks to be executed correctly are relatively sufficient. At present, most researches only focus on the scheduling strategy in the critical level promotion phase, which can make the task more efficient There are few effective schemes for key level security degradation. Therefore, it is necessary to propose a new execution mode and algorithm to achieve the scheduling balance of different levels of tasks.

2. Research Status of MixedCriticality System

The concept of mixed critical system was proposed by vestal at RTSS International Conference on real-time systems field [5], and it was verified that the DM (deadline mononic) priority allocation algorithm used in the traditional task model was no longer optimal. A mixed critical level task model was established, and a fixed priority allocation strategy was proposed based on the OPA (optimal priority assignment) algorithm of Audsley [6]. The white paper [2] written by barhost et al. In 2009 established mixed critical system as an independent research branch. The workshop report at ECRTS2010 predicted the research direction of mixed critical system. After that, the mixed critical system has become a research hotspot in the fields of embedded real-time system and CPS (cyber physical system). Many research institutions, schools and enterprises at home and abroad have set up projects to support the theoretical research and application system design of mixed critical system. mixed critical system has gradually developed into an important branch of real-time system, and the research results are remarkable in recent years.

The task scheduling of mixed critical system needs to meet the following goals: first, it can meet the verification requirements of high critical level tasks when setting relatively conservative time attributes; at the same time, it can meet the design requirements of efficient utilization of processor resources under the optimistic time attribute setting. Baruah et al. [7] proposed a heuristic scheduling algorithm OCBP(own criticality-based priorities) for scheduling limited jobs, and then improved the OCBPalgorithm [8] and proposed LB scheduling algorithm, which can schedule the occasional task set.

At present, the scheduling research based on response time analysis includes: Baruah [9] proposed a response time analysis method suitable for mixed dual critical systems by combining the response time analysis of traditional task model. The method takes the worst-case execution time as the key parameter to judge whether the task is schedulable under pessimistic conditions. After that, the response time analysis was carried out with the period as the key parameter in the literature [10-11]. Reference [12-13] extended the response time calculation method to multi critical level system. In reference [14], a new response time calculation method is proposed, which focuses on the execution time of the task itself when the system is upgraded, and reduces the complexity of the algorithm. Yao et al. [15] assigned different priorities to tasks when the system was at different critical levels, and the priority was assigned by response time analysis combined with OPA algorithm. Burns [16] further proposed a fault-tolerant mixed critical system to ensure the balance between robustness and schedulability.

How to deal with the problem after the key level of the system is upgraded. At present, most scheduling strategies are proposed under relatively conservative time conditions. when the critical level is upgraded, in order to ensure that the high critical level tasks are completed within the deadline, the scheduling mode of sacrificing the low critical level tasks is adopted. This scheduling strategy is based on the off-line pessimistic estimation of system execution, which usually does not conform to the actual application. Therefore, in recent years, researchers have focused on the active processing of tasks with relatively low critical level after the system critical level is upgraded. The hierarchical scheduling algorithm adopted by auderson et al. [17] reduces the impact of high critical level tasks on low critical level tasks through time isolation, and improves the schedulability of low critical level tasks to a certain extent. Burns et al. [18] established a model in which low critical level tasks were executed at a lower service level after the system mode was upgraded. Hang Su [19-20] used ER-EDFalgorithm to allocate the idle time dynamically generated during the execution of high critical tasks to lower critical tasks. The release cycle of low critical level tasks can be adjusted flexibly, and the appropriate idle fragments can be selected for execution. Jan [21] Based on linear programming, low critical level tasks are executed at low frequency after the system critical level is upgraded, thus reducing the system load. Reference [22-23] uses elastic scheduling model to actively deal with low critical level tasks. In reference [24], the concept of "tolerance" is proposed to constrain the condition of critical level promotion and reduce the proportion of low critical level tasks discarded. Ren j et al. [25] used the method of task grouping, high critical level tasks can only affect low critical level tasks in the same group. In addition, the research on how to make the system critical level safe and quickly fall back can also improve the scheduling rate of low critical tasks. Most of the current security degradation schemes are that the system can achieve security degradation in idle time [26-27]. The disadvantage of this method is that the waiting time is long.In order to reduce the fall back time, the literature[28-30] studies the key level security fallback of the system, and puts forward several schemes to realize the security degradation.

Research on dynamic scheduling strategy, multiprocessor scheduling and processor preemption mode includes:S. B and vestal deeply studied the mixed critical level task model [31]. Combined with the OPA algorithm and EDF algorithm proposed by audsly, first use OPA to assign the priority of tasks. If there are multiple tasks that can be assigned a certain priority, then assign the priority according to EDF. In addition, some researchers have studied dynamic adjustment algorithms for mixed critical systems and proposed the concept of virtual deadline dynamic priority scheduling (EDF-VD) [32-35].

In the research of multiprocessor scheduling, the existing task allocation methods can be roughly divided into global scheduling [36] and local scheduling [37]. Auderson et al. [38] discussed the scheduling problem of mixed critical level tasks on multiprocessing platforms for the first time. Then, for the multi critical level model, a hierarchical scheduling algorithm [39] is proposed. Different scheduling algorithms are used for different critical level tasks to ensure that the tasks of different key levels do not affect each other.Li et al. [40] combined with fpEDFalgorithm for traditional real-time systems, studied the scheduling characteristics of incidental tasks. In this paper, we propose two algorithms to modify the cycle: global and global-pragmatic. The paper [41] extended the above algorithm and used partition scheduling method, and proved that local scheduling is better than global scheduling for high priority tasks.Albayati Z et al. [42] proposed a dual partition scheduling algorithm, which based on local scheduling scheme, allows priority low critical tasks to migrate between processors, and improves task partitioning efficiency. Reference [43] proposed a partition strategy for WF-MY, which queries the utilization of all key levels, and adds the utilization of tasks to be allocated into the total utilization of processors, so as to reserve more space for the next task to be allocated and increase the predictability of the partition strategy.Reference [44] studies the local scheduling strategy of multiprocessors. Tasks are allocated to the appropriate processor by "boxing" method, and three allocation strategies are compared, including the first adaptation, the best adaptation and the worst adaptation. R. M. P [45] proposed a response time calculation scheme based on multiprocessor platform and an effective fixed priority allocation algorithm based on OPA. Literature [46-47] studies the change of preemption mode, which reduces the preemption consumption in the process of system execution.

3. Existing Problems and Further Research Directions

Although great achievements have been made in the research ofmixed critical system, it has not met the needs of software and hardware development in actual industry. For example, there is much room for improvement in how to make better use of processor resources and improve the performance of the system.

On the one hand, in order to meet the relatively conservative verification, most scheduling strategies estimate the schedulable ratio offline with the worst case to ensure the smooth execution of high critical level tasks. When the critical level is promoted, the execution time of low critical level tasks is sacrificed, but simply discarding the low critical level tasks may cause data integrity damage and generate some additional consumption. In practical applications, processor resources are usually not fully utilized, and the task execution time is far less than the worst-case execution time. Therefore, the existing scheduling schemes established by static analysis are too negative, and more active scheduling methods need to be further studied.

On the other hand, the focus of the current research on mixed critical system is to ensure the correct execution of tasks in the critical level promotion phase of the system, and the system entering the high critical level mode is a special state. If it is always maintained in this state, it is not in line with the actual situation, so the system also exists in the process of falling from high critical level to low critical level. After the critical level of the system falls down, the suspended low critical tasks can be re executed. It is necessary to further study the key security features of the system to meet the critical requirements of different security levels.

Reference

[1] De Niz D, Lakshmanan K, Rajkumar R. On the scheduling of mixed-criticality real-time task sets[C]//2009 30th IEEE Real-Time Systems Symposium. IEEE, 2009: 291-300.

[2] BarhorstJ , Belote B T , Martin L , et al. White paper：A research agenda formixed-criticalitysystems, http://www.cse.wustl.edu/cdg·ill/CPSWEEK09MCAR:2009-4-30.

[3] Johnson L A. DO-178B, Software considerations in airborne systems and equipment certification[J]. Crosstalk, October, 1998, 199-200.

[4] Palin R, Ward D, Habli I, et al. ISO 26262 safety cases: Compliance and assurance[C]//6th IET International Conference on System Safety 2011. IET, 2011: 1-6.

[5] Vestal S. Preemptive scheduling of multi-criticality systems with varying degrees of execution time assurance[C]//Real-Time Systems Symposium, 2007. RTSS 2007. 28th IEEE International. IEEE, 2007: 239-243.

[6] AudsLey N C. On priority assignment in fixed priority scheduLing.[J]. Information Processing Letters, 2001, 79(1):39-44.

[7] Baruah S, Vestal S. Schedulability analysis of sporadic tasks with multiple criticality specifications[C]//Real-Time Systems, 2008. ECRTS'08. Euromicro Conference on. IEEE, 2008: 147-155.

[8] Li H, Baruah S. An Algorithm for Scheduling Certifiable Mixed-Criticality Sporadic Task Systems[C]// IEEE Real-Time Systems Symposium. IEEE Computer Society, 2010:183-192.

[9] Baruah S K, Burns A, Davis R I. Response-Time Analysis for Mixed Criticality Systems[C]// Real-time Systems Symposium. IEEE Computer Society, 2011:34-43.

[10] Baruah S, Chattopadhyay B. Response-time analysis of mixed criticality systems with pessimistic frequency specification[C]// IEEE, International Conference on Embedded and Real-Time Computing Systems and Applications. IEEE, 2014:237-246.

[11] Zhang N, Xu C, Li J, et al. A sufficient response-time analysis for mixed criticality systems with pessimistic period[J]. Journal of Computational Information Systems, 2015, 11(6):1955-1964.

[12] Fleming T. Extending mixed criticality scheduling[D]. University of York, 2013.

[13] Li L, Li R, Huang L, et al. A New RTA Based Scheduling Algorithm for Mixed-Criticality Systems[C]// IEEE, International Conference on Computational Science and Engineering. IEEE Computer Society, 2013:722-729.

[14] Bai EC, Zhang WZ. Method of response-time analysis for mixed criticality systems. Ruan Jian Xue Bao/Journalof Software, 2015,26(Suppl.(2)):257−262.

[15] Chen Y, Kang G S, Xiong H. Generalizing fixed-priority scheduling for better schedulability in mixed-criticality systems[J]. Information Processing Letters, 2016, 116(8):508-512.

[16] Burns A, Davis R I, Baruah S, et al. Robust mixed-criticality systems[J]. IEEE Transactions on Computers, 2018, 67(10): 1478-1491.

[17] Mollison M S, Erickson J P, Anderson J H, et al. Mixed-Criticality Real-Time Scheduling for Multicore Systems[C]// IEEE International Conference on Computer and Information Technology. IEEE Computer Society, 2010:1864-1871.

[18] Burns A, Baruah S. Towards a more practical model for mixed criticality systems[C]//Workshop on Mixed-Criticality Systems (colocated with RTSS). 2013.

[19] Su H, Zhu D. An Elastic Mixed-Criticality task model and its scheduling algorithm[C]// Design, Automation & Test in Europe Conference & Exhibition. IEEE, 2013:147-152.

[20] Su H, Zhu D, Mosse D. Scheduling algorithms for Elastic Mixed-Criticality tasks in multicore systems[C]// IEEE, International Conference on Embedded and Real-Time Computing Systems and Applications. IEEE, 2013:352-357.

[21] GB/T 7714Jan M, Zaourar L, Pitel M. Maximizing the execution rate of low-criticality tasks in mixed criticality system[J]. Proc. WMC, RTSS, 2013: 43-48.

[22] Huang Lida, li long, Li Renfa, et al. Active processing of low critical level tasks in Mixed critical level multi task scheduling [J]. Computer engineering and science, 2014, 36 (1): 6-11.

[23] Li long. Research on real-time scheduling algorithm for Mixed critical level systems [D]. Hunan University, 2014.

[24] Santy F, George L, Thierry P, et al. Relaxing Mixed-Criticality Scheduling Strictness for Task Sets Scheduled with FP[C]// Real-Time Systems. IEEE, 2012:155-165.

[25] Ren J, Phan L T X. Mixed-Criticality Scheduling on Multiprocessors Using Task Grouping[C]// Euromicro Conference on Real-Time Systems. IEEE Computer Society, 2015:25-34.
[26] Baruah S K , Burns A . Implementing Mixed Criticality Systems in Ada[C]// Reliable Software Technologies-ada-europe -ada-europe International Conference on Reliable Software Technologies. DBLP, 2011.
[27] Tindell K, Alonso A. A very simple protocol for mode changes in priority preemptive systems[R]. Technical report, Universidad Politécnica de Madrid, 1996.
[28] François Santy, Raravi G, Nelissen G, et al. Two protocols to reduce the criticality level of multiprocessor mixed-criticality systems[C]// International Conference on Real-Time Networks and Systems. 2013:183-192.
[29] Bate I, Burns A, Davis R I. A Bailout Protocol for Mixed Criticality Systems[C]// Euromicro Conference on Real-Time Systems. IEEE Computer Society, 2015:259-268.
[30] Bate I , Burns A , Davis R I . An Enhanced Bailout Protocol for Mixed Criticality Embedded Software[J]. IEEE Transactions on Software Engineering, 2017, 43(4):298-320.
[31] Baruah S, Vestal S. Schedulability analysis of sporadic tasks with multiple criticality specifications[C]//Real-Time Systems, 2008. ECRTS'08. Euromicro Conference on. IEEE, 2008: 147-155.
[32] Baruah S, Bonifaci V, D'angelo G, et al. Preemptive uniprocessor scheduling of mixed-criticality sporadic task systems[J]. Journal of the ACM (JACM), 2015, 62(2): 14-47.
[33] Baruah S, Bonifaci V, DAngelo G, et al. The preemptive uniprocessor scheduling of mixed-criticality implicit-deadline sporadic task systems[C]//2012 24th Euromicro Conference on Real-Time Systems. IEEE, 2012: 145-154.
[34] SocciD ,Poplavko P , Bensalem S , et al. Mixed Critical Earliest Deadline First[J]. Euromicro Workshop on Real-Time Systems-Proceedings, 2013, 8114:93-102.
[35] Baruah S. Schedulability Analysis for a General Model of Mixed-Criticality Recurrent Real-Time Tasks[C]// Real-time Systems Symposium. 2017.
[36] Baruah S. Techniques for multiprocessor global schedulability analysis[C]//28th IEEE International Real-Time Systems Symposium (RTSS 2007). IEEE, 2007: 119-128.
[37] Baruah S, Fisher N. The partitioned multiprocessor scheduling of sporadic task systems[C]//26th IEEE International Real-Time Systems Symposium (RTSS'05). IEEE, 2005: 9 pp.-329.
[38] Anderson J H, Baruah S K, Brandenburg B B. Multicore operating-system support for mixed criticality[C]//Proceedings of the Workshop on Mixed Criticality: Roadmap to Evolving UAV Certification. Citeseer, 2009, 4: 7.
[39] Mollison M S, Erickson J P, Anderson J H, et al. Mixed-Criticality Real-Time Scheduling for Multicore Systems[C]// IEEE International Conference on Computer and Information Technology. IEEE Computer Society, 2010:1864-1871.
[40] Li H, Baruah S. Outstanding paper award: Global mixed-criticality scheduling on multiprocessors[C]//Real-Time Systems (ECRTS), 2012 24th Euromicro Conference on. IEEE, 2012: 166-175.
[41] Baruah S, Chattopadhyay B, Li H, et al. Mixed-criticality scheduling on multiprocessors[J]. Real-Time Systems, 2014, 50(1):142-177.
[42] AlbayatiZ , Zhao Q , Youssef A , et al. Enhanced partitioned scheduling of Mixed-Criticality Systems on multicore platforms[C]// Design Automation Conference. IEEE, 2015.
[43] Zhao Qingwei. Task division and implementation in multi-core Mixed criticality real-time system [D]. Huazhong University of science and technology, 2015.
[44] Kelly O R, Aydin H, Zhao B. On partitioned scheduling of fixed-priority mixed-criticality task sets[C]//2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications. IEEE, 2011: 1051-1059.
[45] Pathan R M. Schedulability Analysis of Mixed-Criticality Systems on Multiprocessors[C]// Euromicro Conference on Real-Time Systems. IEEE Computer Society, 2012:309-320.
[46] Zhao Q, Gu Z, Zeng H. PT-AMC: Integrating Preemption Thresholds into Mixed-Criticality Scheduling[C]// Design, Automation & Test in Europe Conference & Exhibition. 2013.
[47] Burns A, Davis R I. Adaptive Mixed Criticality Scheduling with Deferred Preemption[C]// Real-time Systems Symposium. 2014.