

Statistical analysis of execution time profile for temporal validation of a distributed hard real-time system

Arpitha Prabhakara
Kompetenzzentrum Automobil und
Industrie Elektronik GmbH
Villach, Austria
Arpitha.Prabhakara@k-ai.at

Benjamin Steinwender
Kompetenzzentrum Automobil und
Industrie Elektronik GmbH
Villach, Austria

Wilfried Elmenreich
Networked and Embedded Systems
Alpen-Adria-Universität
Klagenfurt, Austria
Wilfried.Elmenreich@aau.at

Abstract—This paper addresses modelling and assessing of worst-case execution times of tasks in a distributed real-time system. The presented case study is based on a stress test system for power semiconductors.

Typical distributed systems involve inherent inter-dependencies that have to be handled with minimized delays. Studying the computing task's timing behavior becomes necessary to understand the timing violations. Modelling the temporal behaviour of the system using probabilistic Worst-Case Execution Time (WCET) analysis aids to overcome the timing faults.

This paper focuses on (1) the study of the temporal behaviour through the execution time profile of the distributed hard real-time system; (2) the statistical analysis by performing probability distribution modelling on the measured data. Measurement-based probabilistic timing analysis is an emerging and reliable method used to arrive at industry quality estimates. This method is used here in the paper to carry out temporal validation of the real-time computing tasks on our case study.

Index Terms—Real-Time Distributed System, Timing Failure, WCET Analysis, Measurement-based Probabilistic Timing Analysis, Statistical Analysis, Probability Distribution, Fault-Tolerant System

I. INTRODUCTION

To ensure the proper operation of power semiconductors over their lifetime (which in the example of photovoltaic applications can be up to 25 years), it is necessary to assess their durability via electro-thermal stress testing [1]–[3]. This has given rise to the development of application-oriented stress test systems accommodating various test specifications to test the individual components [4]–[6]. Such test systems are physical systems testing power electronic components using high voltages (more than 1 kV) and/or high currents (100 A and above). Furthermore, the test system is a distributed system consisting of multiple nodes that produce the results in a stipulated timeline. Errors or timing violations result in catastrophic failure, significant property damage, or may endanger a human's life. Hence, the results generated should be correct and valid within a bounded timeline, thus establishing the requirements for a distributed hard real-time system. Alongside, the system should be able to respond within the time frame of the functional tasks even with the anticipated error conditions [7].

Distribution, fault tolerance and real-time performance are the characteristics of a responsive system [8].

The temporal properties play a very crucial role in the performance of the complex hard real-time system [8]. The operational correctness of a real-time system depends on the timely completion of the underlying tasks [9]. To analyse such a system, it is necessary to know the response time of the individual tasks [10]. This can be achieved by WCET analysis, which determines the upper bound for the execution time of the individual tasks [11].

A high quality estimation of WCET is challenging as the cost-effectiveness consideration is required for industry [12]. The two main identified challenges are execution-time modelling of the hardware and the path problem that forbids capturing the WCET by end-to-end measurements due to limits in computational complexity [13].

Within the power electronic test, devices operate with high switching frequency and repetitive power surges. The communication interface should be highly reliable in hard real-time systems. In order to achieve reliability, the communication interface must provide data transfer that can withstand global faults and local permanent faults [8]. In [14], the time-triggered (TT) and the event-triggered (ET) approach are compared with respect to the properties like temporal behaviour, predictability issues, resource usage and assumption coverage. A TT system has increased design effort, whereas easier verification of temporal correctness is achieved. In the case of ET systems, predictions about their temporal behaviour are more difficult.

The conditioning signals, temporal, and application dependencies need to be controlled to carry out the testing smoothly. In this paper, the test systems investigated are distributed embedded systems that combine a TT and ET approach. This gives rise to temporal dependencies and intercommunication dependencies between the entities, requiring a proper assessment of a task's execution time. The contribution of this paper is to elaborate an execution time model of the depicted distributed hard real-time system based on measurements and statistic modeling.

II. RELATED WORK

A fault in a distributed real-time system occurs when the system fails to deliver the actual intended output before a given deadline. The faults can occur due to timing failure, response failure, network and nodal failure [8].

A timing failure occurs if a system delivers a correct value at the wrong time. This can occur too early or too late. The knowledge of the upper bound of the execution time of a task is necessary to guarantee timeliness of the system. Therefore, it is required to estimate WCET to guarantee timing validation [15]. A reliable analysis of WCET is quite challenging with respect to the nondeterministic performance of the complex hardware and software [16]. The embedded system's complexity has led to extensive research in the area of timing analysis.

The WCET depends on the timing properties of the hardware and programming code logic. The main goal of WCET is to obtain the maximum execution time of a given code segment for a specific run-time platform [17]. The approaches in the estimation of WCET has below mentioned methods.

Static methods focus on analysing the possible control flow paths, combining them with the abstract model of hardware, and obtaining an upper bound. This does not rely on executing code on a simulator or real hardware [12].

Measurement-based probabilistic timing analysis are widely explored and industrialized. The jitter sources that affect the timing behaviour of the program execution are to be considered. Here, the probabilistic bound on program execution is derived through analysis-time observation [18].

Statistical modelling provides the knowledge on the system's temporal behaviour to check the hypothesis made on the timing analysis. Statistical probability analysis has emerged to be highly reliable with system complexity and cost effective in industrial domain. Probabilistic modelling seeks a priori guarantees where the time resources used in the execution platform answers the probabilistic reasoning [18].

According to [12], the path-subset challenge for timing analysis can be handled by measuring each code segment. However, it is still not possible to eliminate the unsafe upper bounds. Furthermore, it is expensive to carry out testing of all execution paths. The probabilistic approach used for WCET analysis results in a probability density function for WCET instead of single value description. In [19], the approach has been tested on an automotive test-bed with a microcontroller.

The highest priority activities usually have timeliness guarantees and others eventually will have a bound which is far from their typical execution time. Thus, instead of having a step function as the distribution function for the execution time we tend to get a distribution as depicted in Figure 1.

Casimiro *et al.* [21] propose the so-called timely computing base where the duration is measured. The timely execution is verified by estimating WCET [22] and then followed by timing failure detection. The timing failure detection proposed in the above paper covers the completeness and the accuracy. There is a distinction between methods preventing errors that causes a timing failure and the means to detect and tolerate timing failures.

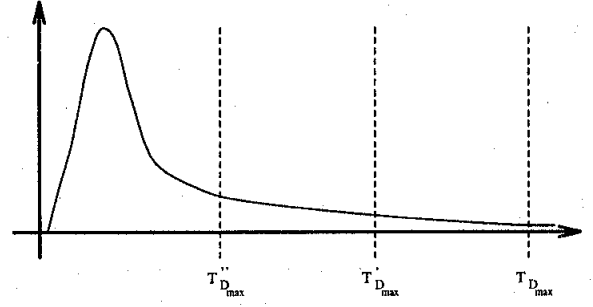


Figure 1. Distribution function for low-priority message delivery time (Example from [20])

Almeida [20] deals with partially synchronous models assuming synchronization with global time. The paper proposes using timing failure detection service by group communication to achieve safety in a timely fashion. The dynamic system with no completely controlled load is prone to timing faults. Thus, it is difficult to know the bounds for all the activities and evaluate WCET in such an environment cost-effectively.

Accordingly, mentioned in [14], in a distributed system the failure of a single node is equivalent to the whole system's failure. ET communication hardly provides any guarantee of timeliness. Time-triggered architecture (TTA) is widely used in avionics and automotive domains for safety-critical applications [23].

III. CASE STUDY

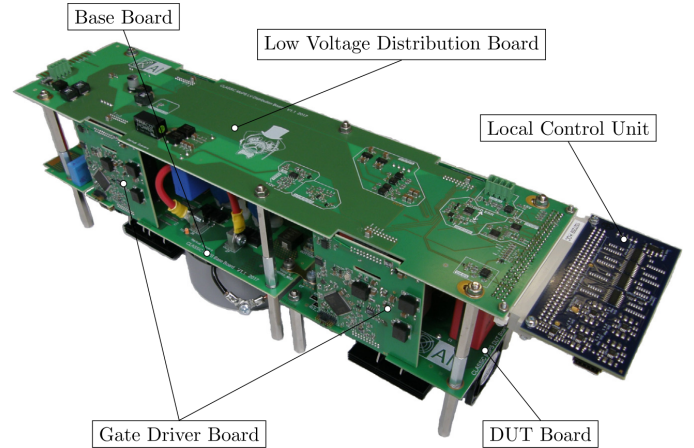


Figure 2. Distributed Real-Time System: A Case Study

Our case study (Figure 2) refers to a power electronic application for testing discrete power semiconductors [24]. The power application is controlled by a distributed hard real-time system consisting of a host computer communicating with multiple application control nodes. The host controls different external peripherals (like power supplies) and stores the measurements and result data, received from the nodes, in a database. These nodes act as real-time entities and communicate with the host

computer via Ethernet. During the power application operation, a subsystem of several ARM Cortex-M4 microcontroller nodes carries out real-time control, data acquisition, and protection functions. The microcontroller nodes have TT communication with the real-time entities. Subsequently, the host to the application nodes follows ET communication. The control nodes' software execution tasks have event execution, interrupt handling, and polling mechanisms. These tasks give rise to mutual interaction, which necessitates working together with a given timing.

The major advantage of the concept described in Figure 2 is that the measurements are directly performed on the application control node. There is event-triggered communication between the node and the host, whereas time-triggered communication is handled between real-time entities. Time synchronization is achieved between nodes and the host through a distributed synchronization algorithm.

IV. PROBLEM STATEMENT AND OBJECTIVES

There can be loss of information based on severe faults like hardware malfunction. Furthermore, the control nodes are in vicinity of the power application where switching noise affects the communication. Consequently, the system becomes vulnerable to temporal failures since the currently active node would wait for the missing message. Due to these error scenarios, the bus communication and the real-time system would stop operating. As the set-up deals with high power, this issue needs to be tackled to accommodate the system to the desired fail-silent state.

In this research, we deal with the timing violations that lead to the erroneous nodes causing temporal faults and resulting in real-time communication failures. The faults that affect the distributed hard real-time system's temporal behavior and the analysis methods to aid in determining the above-mentioned faults have to be investigated. Measurement-based timing analysis is used for execution-time modelling of the hardware by estimating the WCET [17].

To detect and handle the timing faults, performance modelling aims to ensure the system meets its target needs. Timing fault tolerance solution should be able to: 1) measure the timing duration 2) detect timing failures in an accurate fashion 3) guarantee timely action to handle failures [25].

V. METHODOLOGY

In our case study, a measurement campaign is carried out by identifying the software tasks that constitute the system's real time functionality. This campaign includes measuring the execution time of mentioned functions tasks running on the application control node. The execution time has been captured for repetitive samples. Based on the measurement samples, the histogram of the samples is plotted for the frequency of occurrences. The probability distribution function is calculated and curve fitting is done to identify the best fit. The identified distribution is used to provide a statistical estimate of the WCET bound of the task [12].

The execution time provides us the execution time profile of the application control node's functions. This aids us in temporal validation of the functions by providing the characteristics and behaviour of the frequency of occurrences of the function. Execution time profile (ETP) is used to conduct statistical analysis, which derives the worst case execution time estimation. This eliminates the need to derive an accurate actual model of the hardware.

VI. RESULTS & DISCUSSIONS

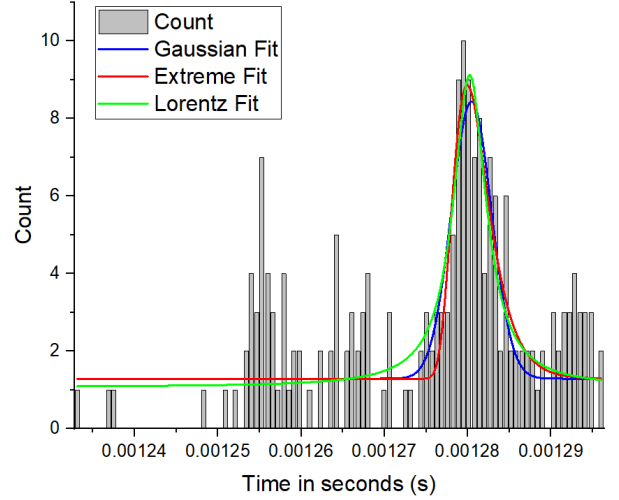


Figure 3. Task 1 execution time histogram and probability distribution fit

Table I
DESCRIPTIVE STATISTICS OF TASK 1

| | N Total | Mean | Std.deviation | Min.value | Max.value |
|------|---------|---------|---------------|-----------|-----------|
| Time | 100 | 1.26 ms | 18.45 μ s | 1.20 ms | 1.30 ms |

The measurement campaign is carried out by measuring the application control node's data acquisition code. Task 1 handles returning the operational result from the real-time entities. The measured data is shown in Figure 3 which provides the execution time profile for the system's functions. The descriptive statistics values for Task 1 are shown in Table I.

The measurement is carried out to obtain 200 samples. A processing step on the raw data is carried out to obtain the maximum and minimum values for the bin. The histogram is plotted in Figure 2 with bin size and sample size of 100. The plot describes the relation between the execution time to their relative occurrence frequency. The distribution fitting is performed to determine the best fit by performing a chi-square test. Gauss, General extreme value, and Lorentz distributions were among the best fit. With the chi-square value obtained and degree of freedom, the General Extreme Value distribution tops the fit. Hence, the expected bin values within the best fit distribution will be accounted for in the estimation.

Task 2 returns the analog measurement values from the real-time entities. The measurement samples were captured for

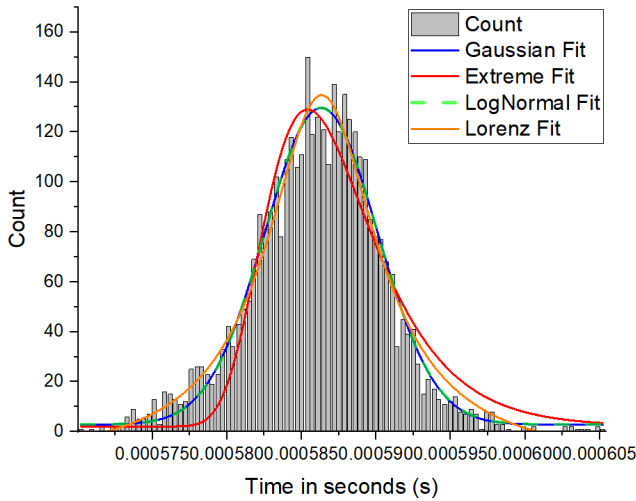


Figure 4. Task 2 execution time histogram and probability distribution fit

Table II
DESCRIPTIVE STATISTICS OF TASK 2

| | N Total | Mean | Std.deviation | Min.value | Max.value |
|------|---------|----------------|---------------|----------------|----------------|
| Time | 100 | 587.69 μ s | 10.28 μ s | 570.20 μ s | 605.20 μ s |

lower sets. This did not produce enough data to carry out the distribution fitting. Therefore, a higher set of 3500 samples were captured.

To eliminate the ambiguity to obtain the execution time, measurement for 3500 samples was carried out. Table II shows the descriptive statistics of Task 2. In Figure 4, frequency of occurrence versus the execution time histogram bins are plotted. Chi-squared test performed to obtain the best fit resulted in Gauss, General extreme value, Lognormal and Lorentz distributions. The chi-squared value corresponding to the degree of freedom shows that Gaussian fit is the best fit out of all. This will be considered for a further statistical estimate to derive worst-case execution timing.

VII. CONCLUSION & OUTLOOK

According to the literature, the path-based execution bound needs instrumentation for program segmentation timing analysis. This additionally adds overhead and the analysed code will differ from the actual code which affects the precise upper bounds [13]. Measurement-based probabilistic timing analysis are considered at first place for the industrial domain whereas the limitations exist in lack of firmware support and testing on the applications [18]. The challenge exists in supporting the hypotheses for measurement-based timing analysis on the grounds of coverage obtained on program paths for execution time. The problem that the longest execution path might be overlooked in a measurement-based approach still exists. The idea in this paper is that: 1) many embedded system tasks are less complex regarding their execution branch. Therefore above problem is a lesser concern; 2) modern processor architectures make a proper static analysis very difficult due to different

clocking modes and caching conflicts. A measurement-based approach is thus an important contribution towards validation of the system. Hence, a measurement-based probability timing analysis was selected as a minimal hybrid approach for the given problem.

The measurements carried out support us in providing temporal validation and execution time profile, which is then utilized for WCET estimation. In this paper, we have elaborated a measurement of the tasks' execution time and determined a probabilistic execution timing model through probability distribution fitting.

To carry out the timing analysis, we did tests with a minor injection of measurement code as well as tests where the code under test was not modified at all. This maintains the authenticity of the code and refrains it from any additional overhead instrumentation. This drawback with the WCET experiments on the industrial hardware is taken care in this paper. The estimated results from this investigation will contribute to the legitimate temporal definiteness.

VIII. FUTURE WORK

The presented approach allows deriving a WCET estimate using measurements. The next steps will be to discuss the used methods used, and, in particular, to investigate the validity of applying the extreme value distribution as suggested by our results. A reasonable upper bound estimation will be used as the input to the developed scheduler tool to ensure temporal correctness of the system.

ACKNOWLEDGMENT

This work was funded by the Austrian Research Promotion Agency (FFG, Project No. 881110). I would like to express my gratitude to the colleagues for their valuable discussions and feedback which contributed towards this paper.

REFERENCES

- [1] F. Blaabjerg, K. Ma, and D. Zhou, "Power electronics and reliability in renewable energy systems," in *Industrial Electronics (ISIE), 2012 IEEE International Symposium on*, IEEE, 2012, pp. 19–30.
- [2] R. Sleik, M. Glavanovics, S. Einspieler, A. Muetze, and K. Krischan, "Modular test system architecture for device, circuit and system level reliability testing and condition monitoring," *IEEE Transactions on Industry Applications*, vol. 53, pp. 5698–5708, 6 Nov. 2017, ISSN: 0093-9994. DOI: 10.1109/TIA.2017.2724501.
- [3] U. Choi, F. Blaabjerg, and S. Jørgensen, "Power cycling test methods for reliability assessment of power device modules in respect to temperature stress," *IEEE Transactions on Power Electronics*, vol. 33, no. 3, pp. 2531–2551, Mar. 2018, ISSN: 1941-0107. DOI: 10.1109/TPEL.2017.2690500.

- [4] P. Ghimire, A. R. de Vega, S. Bęczkowski, B. Rannestad, S. Munk-Nielsen, and P. B. Thøgersen, "Improving Power Converter Reliability: Online Monitoring of High-Power IGBT Modules," *IEEE Industrial Electronics Magazine*, vol. 8, no. 3, pp. 40–50, Sep. 2014, ISSN: 1932-4529. DOI: 10.1109/MIE.2014.2311829.
- [5] S. Pu, E. Ugur, and B. Akin, "Real-time degradation monitoring of SiC-MOSFETs through readily available system microcontroller," in *2017 IEEE 5th Workshop on Wide Bandgap Power Devices and Applications (WiPDA)*, Oct. 2017, pp. 378–382. DOI: 10.1109/WiPDA.2017.8170576.
- [6] F. Erturk, E. Ugur, J. Olson, and B. Akin, "Real-Time Aging Detection of SiC MOSFETs," *IEEE Transactions on Industry Applications*, vol. 55, no. 1, pp. 600–609, Jan. 2019, ISSN: 0093-9994. DOI: 10.1109/TIA.2018.2867820.
- [7] R. Ramezani and Y. Sedaghat, "An Overview of Fault Tolerance Techniques for Real-Time Operating Systems," in *3rd International eConference on Computer and Knowledge Engineering*, Mashhad, Iran: 3th International eConference on Computer and Knowledge Engineering (ICCKE 2013), Feb. 2013. DOI: 10.1109/ICCKE.2013.6739552.
- [8] H. Kopetz, *Real-time systems: design principles for distributed embedded applications*. Springer Science & Business Media, 2011.
- [9] Y. Zhang and K. Chakrabarty, "Fault recovery based on checkpointing for hard real-time embedded systems," in *Proceedings of the 18th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, IEEE, 2003.
- [10] Y. Lu, "Approximation Techniques For Timing Analysis Of Complex Real Time Embedded Ssystems," Ph.D. dissertation, Maelardalen University of Sweeden, 2002.
- [11] T. Kelter, "WCET Analysis and Optimization for Multi-Core Real-Time Systems," Ph.D. dissertation, Technischen Universitaet Dortmund, 2015.
- [12] R. Wilhelm, T. Mitra, F. Mueller, I. Puaut, P. Puschner, J. Staschulat, P. Stenström, J. Engblom, A. Ermedahl, N. Holsti, S. Thesing, D. Whalley, G. Bernat, C. Ferdinand, and R. Heckmann, "The worst-case execution-time problem—overview of methods and survey of tools," *ACM Transactions on Embedded Computing Systems*, vol. 7, no. 3, pp. 1–53, Apr. 2008. DOI: 10.1145/1347375.1347389.
- [13] R. Kirner and P. Puschner, "Obstacles in Worst-Case Execution Time Analysis," in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, IEEE, May 2008, pp. 333–339, ISBN: 978-0-7695-3132-8. DOI: 10.1109/ISORC.2008.65.
- [14] W. Elmenreich, "Sensor Fusion in Time-Triggered Systems," Ph.D. dissertation, Vienna University of Technology, 2002.
- [15] C. Ferdinand, R. Heckmann, M. Langenbach, F. Martin, M. Schmidt, H. Theiling, S. Thesing, and R. Wilhelm, "Reliable and Precise WCET Determination for a Real-Life Processor," AbsInt Angewandte Informatik GmbH, Saarbruecken, Fachrichtung Informatik, Universitaet des Saarlandes, Saarbruecken, Tech. Rep. LNCS 2211, 2001, pp. 469–485.
- [16] S. Einspieler, B. Steinwender, and W. Elmenreich, "Integrating Time-Triggered and Event-Triggered Traffic in a Hard Real-Time System," in *1st IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, May 2018, pp. 122–128. DOI: 10.1109/ICPHYS.2018.8387647.
- [17] R. Kirner and P. Puschner, "Classification of WCET Analysis Techniques," in *Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC'05)*, IEEE, 2005, pp. 190–199. DOI: 10.1109/ISORC.2005.19.
- [18] F. Cazorla, L. Kosmidis, C. Hernandez, J. Abella, and T. Vardanega, "Probabilistic worst-case timing analysis taxonomy and comprehensive survey," *ACM Computing Surveys Publication*, vol. 52, no. 1, p. 35, Feb. 2019. DOI: 10.1145/3301283.
- [19] G. Bernat, A. Colin, and S. M. Petters, "WCET analysis of probabilistic hard real-time systems," in *23rd IEEE Real-Time Systems Symposium, 2002. RTSS 2002*, IEEE, Dec. 2002. DOI: 10.1109/REAL.2002.1181582.
- [20] C. Almeida, "Timing failure detection and real-time group communication in quasi-synchronous systems," in *Proceedings of EURWRTS*, IEEE, 1996.
- [21] A. Casimiro, P. Martins, P. Verissimo, and L. Rodrigues, "Measuring distributed durations with stable errors," in *Proceedings of the 22nd IEEE Real-Time Systems Symposium*, London, UK: IEEE, 2001, pp. 310–319. DOI: 10.1109/REAL.2001.990630.
- [22] H. Kopetz, "Event-triggered versus time-triggered real-time systems," in *Proceedings of the International Workshop on Operating Systems of the 90s and Beyond*, London, UK, UK: Springer-Verlag, 1991, pp. 87–101.
- [23] H. Kopetz, G. Bauer, and S. Poledna, "Tolerating arbitrary node failures in the time-triggered architecture," SAE Technical Paper, Tech. Rep., 2001. DOI: 10.4271/2001-01-0677.
- [24] M. Sievers, M. Glavanovics, C. Rhinow, and B. Findenig, "Modular application relevant stress testing for next generation power semiconductors," *Microelectronics Reliability*, vol. 100–101, p. 113 330, 2019, ISSN: 0026-2714. DOI: 10.1016/j.microrel.2019.06.022.
- [25] A. Casimiro and P. Verissimo, "Generic Timing Fault Tolerance using a Timely Computing Base," in *Proceedings of the International Conference on Dependable Systems and Networks*, IEEE, 2002.