Modeling and schedulability analysis of AFDX networks in MAST 2

J. Javier Gutiérrez

Computers and Real-Time Group, University of Cantabria, 39005-Santander, SPAIN
gutierjj@unican.es

EXTENDED ABSTRACT[1]

This short paper reports an overview of MAST (Modeling and Analysis Suite for real-Time applications) [1], and how AFDX networks will be modeled and analyzed within this framework. MAST defines a model for describing the timing behavior of distributed real-time systems and also includes a set of tools for schedulability analysis, assigning scheduling parameters and performing sensitivity analysis. It is developed by the Computers and Real-Time Group at the University of Cantabria, and it has been conceived for research purposes as a long term project led by Michael González Harbour. The objective of this project is to propose an open model as a basis to deal with new needs for timing behavior modeling and as a work bench for future timing analysis and optimization techniques.

The MAST model [2] is now aligned with the OMG MARTE standard [3], especially with the SAM profile (Schedulability Analysis Modeling). MAST defines a high-level model mainly consisting of the following basic elements: *Execution Platform* (CPUs and communication networks), *Schedulable Resources* or *Scheduling Servers* (tasks or message streams), *Operations* (code blocks or messages), *Mutual Exclusion Resources* or *Shared Resources* (resources that must be used in a mutually exclusive way), and *End-to-End Flows* (key elements that will be described later). It also has a rich overhead model for other elements such as *Timers* or *Network Drivers*, and also a high expressiveness for timing requirements. This high-level model is transformed into analysis or simulation models over which schedulability analysis techniques or simulation tools can be applied.

The analysis model considers a system composed of distributed *end-to-end flows*, each released by a periodic or sporadic sequence of *external events*, and containing a set of *steps* that model tasks and messages. Each release of an end-to-end flow causes the execution of the set of steps, each step being released when the preceding one in its end-to-end flow finishes its execution. We assume that all event sequences that arrive at the system and their worst-case rates are known in advance, and we also assume that tasks and messages are statically allocated in processors and networks. The relative phasing of the activations of different end-to-end flows is assumed to be arbitrary. Messages and communication networks can be treated in a similar way as tasks in processing resources.

Each step of an end-to-end flow has a worst and best-case execution times, and can have a global deadline referred to the activation of the external event (end-to-end deadline for the final step) or a local deadline referred to the activation of the step itself. We allow deadlines to be larger than the periods. As a result of the schedulability analysis, each step also has a worst-case response time (or an upper bound of it) and a best-case response time (or a lower bound of it). The worst-case response times can be compared with the deadlines in order to determine the schedulability of the system. Our model takes into account the maximum release jitter of the event that triggers each step in the end-to-end flow, and can also manage an initial offset, which is the minimum release time for a step, referred to the activation of the external event that triggers its end-to-end flow. We assume that jitters and offsets may be larger than periods.

This end-to-end flow model allows the application of different analysis techniques for Fixed Priorities (FP) [4], and Earliest Deadline First (EDF) with or without clock synchronization [5][6]. These techniques can also be combined in heterogeneous systems where some processing resources are scheduled with FP and the others with EDF [7]. They are based on the so-called holistic response-time analysis, where all the steps of an end-to-end flow are considered independent of the others, except for the variability introduced by the release jitter, or the precedence relations represented by offsets. Although all these analyses are pessimistic, part of this pessimism can be removed by offset-based techniques [8][9].

In this context, MAST 2 [10] adds the modeling of AFDX (Avionics Full Duplex Switched Ethernet) [11], which is a communication network based on the use of point-to-point full-duplex Ethernet links and special purpose

1. This work has been funded in part by the Spanish Government and FEDER funds under grant number TIN2011-28567-C03-02 (HI-PARTES).

switches in which the routing of messages is statically preconfigured. Traffic regulation in AFDX is made at the send endpoints via *Virtual Links*, which establish logical unidirectional connections from one source end system to one or more destination end systems, crossing one or more switches, and having a dedicated maximum bandwidth controlled by two parameters: Lmax (the largest Ethernet frame in bytes), and GAP (the Bandwidth Allocation Gap, which is a minimum interval in milliseconds between Ethernet frames transmitted on a virtual link). Each virtual link has a FIFO queue and can be shared by up to four *Sub-Virtual Links*, each of them having a dedicated FIFO queue which is read on a round-robin basis by the virtual link. The AFDX switches deliver messages from an incoming port to an outgoing port or ports in a store-and-forward way. Messages are enqueued at outgoing ports in two prioritized FIFO queues (high and low priorities), which can be configured on a virtual link basis.

To deal with AFDX networks, MAST 2 [10] now supports new modeling elements:

- *Network Switches* and *Routers*. An *AFDX Switch* is defined to work according to the AFDX specification.
- *AFDX Links*. Links between a processor and/or a switch in a network that uses the AFDX protocol, allowing full duplex communications.
- *Communication Channels*. Specialized schedulable resources for message transmission in a network. A default communication channel is predefined for each AFDX Link with an implicit FIFO scheduling policy.
- *AFDX Policy*. The scheduling policy used by AFDX, in which messages are scheduled through virtual links.
- *AFDX Virtual Links*. To support the scheduling parameters applicable to schedulers with the AFDX Policy.

We have also developed a response time analysis technique for AFDX networks [12] which can be integrated with the analysis of the processors in the holistic approach for heterogeneous distributed systems presented in [7]. This technique has the following significant characteristics:

- It can handle messages with arbitrary message periods and release jitter.
- It allows the analysis of virtual links and also sub-virtual links.
- Multi-packet messages can be analyzed, thus obtaining response times for the complete messages.
- It can handle priorities in the switches.
- Although this technique is pessimistic, it is scalable and it can be applied to real systems with more than 1000 virtual links.

Other techniques can be found in the literature for response-time analysis of AFDX networks, but most of them consider the network in isolation and obtain similar results for a more simplified model (see the related work section in [12]).

The MAST 2 model is not yet available in the MAST suite of tools, so a standalone prototype tool of the AFDX analysis has been made; we plan to integrate this new technique with MAST 2 in the near future. MAST 2 also allows modeling partitioned systems (with a new *Timetable Driven Policy*), and we have developed a response-time analysis technique for these kind of systems that will be published soon. This will enable modeling and analyzing full ARINC-653-based distributed systems [13], such as those used in avionics.

References

[1] MAST home page: http://mast.unican.es

[2] M. González Harbour, J. Javier Gutiérrez, J.C. Palencia, and J.M. Drake, "MAST: Modeling and Analysis Suite for Real Time Applications," Proc. of the 13th Euromicro Conference on Real-Time Systems, Delft (The Netherlands), pp. 125-134, 2001.

[3] Object Management Group, "UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded systems," 2009 OMG Document, v1.0 formal/2009-11-02.

[4] K. Tindell, and J. Clark, "Holistic Schedulability Analysis for Distributed Hard Real-Time Systems," Microprocessing & Microprogramming, Vol. 40, Nos.2-3, pp. 117-134, 1994.

[5] M. Spuri, "Holistic Analysis for Deadline Scheduled Real-Time Distributed Systems," Tech. Rep. RR-2873, INRIA, France, April 1996.

[6] Juan M. Rivas, J. J. Gutiérrez, J. C. Palencia, and M. González Harbour, "Optimized Deadline Assignment and Schedulability Analysis for Distributed Real-Time Systems with Local EDF Scheduling," Proc. of the 8th International Conference on Embedded Systems and Applications, Las Vegas (Nevada), USA, 2010.

[7] Juan M. Rivas, J. Javier Gutiérrez, J. C. Palencia, and M. Gonzalez Harbour, "Schedulability Analysis and Optimization of Heterogeneous EDF and FP Distributed Real-Time Systems," Proc. of the 23th Euromicro Conference on Real-Time Systems, Porto (Portugal), 2011.

[8] J.C. Palencia, and M. González Harbour, "Schedulability Analysis for Tasks with Static and Dynamic Offsets," Proc. of the 19th Real-Time Systems Symposium, IEEE, pp. 26-37, 1998.

[9] J. Mäki-Turja, and M. Nolin, "Efficient implementation of tight response-times for tasks with offsets," Real-Time Systems Journal, 40(1), pp. 77-116, 2008.

[10] M. González Harbour, J. Javier Gutiérrez, José M. Drake, P. López Martínez, and J. C. Palencia, "Modeling distributed real-time systems with MAST 2," Journal of Systems Architecture, Vol 56(6), pp. 331-340, Elsevier, 2013.

[11] Airlines Electronic Engineering Committee, Aeronautical Radio INC., "ARINC Specification 664P7: Aircraft Data Network, Part 7 - Avionics Full Duplex Switched Ethernet (AFDX) Network," June 27, 2005.

[12] J. Javier Gutiérrez, J. C. Palencia, and M. González Harbour, "Holistic schedulability analysis for multipacket messages in AFDX networks," In press, Real-Time Systems Journal, 2013, DOI:10.1007/s11241-013-9192-2.

[13] ARINC, Avionics Application Software Standard Interface, ARINC specification 653-1, March 2006.