## Methodology of Combining Empirical Stress Testing and Formal-Methods Based Schedulability Analysis for Real-Time Multicore Software

Carnegie Mellon University: Bjorn Andersson, Dionisio de Niz

OCTOBER 2023

© 2023 Carnegie Mellon University

D i stribution A: Approved for Public Release. D i stribution is Unlimited Carnegie Mellon University Software Engineering Institute Copyright 2023 Carnegie Mellon University, U.S. Army Combat Capabilities Development Command Aviation & Missile Center.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

DM23-0820

## The need for safety (of aircraft)

<u>Ariane 5 Flight 501:</u> Catastrophic failure (causing self-destruct) in flight in 1996 due to incorrect software reuse.

**Boeing 737 MAX:** Two crashes (in 2018 and 2019) due to software flaw in the MCAS (Maneuvering Characteristics Augmentation System).

**<u>Airbus A400M</u>**: Crash during a test flight in Spain 2015 due to software flaw.

Helicopters: Army Helicopter crashes in April 2023: Apache in Alaska and Black Hawk in Kentucky.

Some crashes are caused by software. The increase complexity of software will make the impact of software on safety more important.









### The need for speed (of software development of aircraft)



#### The Department will instead reward rapid experimentation, acquisition, and fielding.











Even if we decrease the development time to zero, then certification time is still there and becomes a bottleneck for our ability to field new systems rapidly.



Test data as cert evidence

- Exhaustive takes too long
- Non-exhaustive is unsafe

Formal methods have the potential to avoid these

#### Formal methods



Input: (i) a model of a system and (ii) a correctness condition.

#### Output: True/False/Undecided

## Different correctness properties

Logical correctness: Compute 2+3. The result should be 5.

Temporal correctness:

Compute 2+3.

The time from when the computation is requested until its result delivered should be at most 20 milliseconds.



Focus of this talk

#### Why timing of software matter? Computer Program Physical Environment Actuator Computer Program Computer Program Computer Physical Environment Computer Physical Environ

Interaction of the software with the physical environment imposes timing requirements on software.

"Timing problems are one of the common causes of run-time failures in process-control systems, and timing is often inadequately specified."

M.S. Jaffe, N.G. Leveson, M.P.E. Heimdahl, and B.E. Melhart, "Software requirements analysis for real-time process-control systems," IEEE TSE, 1991.

### Why timing of software matter?

Army avionics interact with physical environment. The world does not stand still.

"The trick there, when you're processing flight critical information, it has to be a deterministic environment, meaning we know exactly where a piece of data is going to be exactly when we need to — no room for error," Langhout says. "On a multi-core processor there's a lot of sharing going on across the cores, so right now we're not able to do that." - Jeff Langhout, Acting Director, U.S. Army Aviation and Missile Research Development and Engineering Center (AMRDEC)

Source: "Army still working on multi-core processor for UH-60V," May 2017, Available at https://www.flightglobal.com/news/articles/army-still-working-on-multi-core-processor-for-uh-6-436895/.

## Why satisfying timing requirements is challenging?

Satisfy for all scenarios

Depends on underlying hardware platform—not just software

Depends on external physical world

Event-driven

Undocumented hardware

#### Multitasking, Inter-core interference

```
int thread1() {
    perform_initialization_thread_1();
    while (1) {
        wait_for_event_thread1();
        s = read_sensor_thread1();
        o = perform_processing_thread1(s);
        actuate_command_thread1(o);
    }
}
```

```
int thread1() {
  perform initialization thread 1();
  while (1) {
     wait for event thread1();
     s = read sensor thread1();
     lock semaphore(sem);
     d = read shared data thread1();
     unlock semaphore(sem);
     o = perform processing thread1(s,d);
     actuate command thread1 (o);
```

```
int thread1() {
       perform initialization thread 1();
       while (1) {
         wait for event thread1();
         s = read sensor thread1();
         lock semaphore(sem);
deadline
         d = read shared data thread1();
         unlock semaphore(sem);
         o = perform processing thread1(s,d);
         actuate command thread1 (o);
```

```
int thread1()
       perform initialization thread 1();
       while (1) {
         wait for event thread1();
         s = read sensor thread1();
         lock semaphore(sem);
deadline
         d = read shared data thread1();
         unlock semaphore(sem);
         o = perform processing thread1(s,d);
         actuate command thread1 (o);
```

Many sources of delay:

```
Thread's own execution
```

The thread may get preempted The thread may block on semaphore Thread may experience inter-core interference (multicore) Q: How to verify timing of software executing on multicore?

A: Use a two-step framework where some activities may use formal methods and some may not.

Well-known in the research literature

Focus of this talk

Carnegie Mellon University Software Engineerin Institute



Carnegie Mellon University Software Engineeriny Institute



## Are there tools for these activities?



## Read design documents and source code.



## Are there tools for these activities?



# Rich literature. Let us see some of SEI's and AvMC's tools.



Carnegie Mellon University Software Engineering Institute

# Single-core. Find Estimate of Worst-Case Execution Time.



## Single-core. Find Estimate of Worst-Case Execution Time.

Find resource characteristics of each thread/process	
Find timing requirements of each thread/process	Make sure that all threads/processes satisf timing requirements whe executing together on a shared computer platform
Find invocation pattern of	

### Worst-Case Execution Time Analysis

	ba@ba-desktop:/ga_find_wcet	
File El	Edit View Search Terminal Help	
baijba-	<pre>i-desktop:-/ga_find_wcet\$ more myconf</pre>	
2		
262144	4	
bayba- 0.3063	<pre>-desktop:-/ga_find_wcetS ./ga_find_wcet myconf ./bubbleso 3359</pre>	rt
bagba-	-desktop:-/ga_find_wcet\$ time ./bubblesort < ascending_in	tegers.dat
real	0n0.0025	
user	0n0.001s	
sys	0n0.0015	
bağba-	-desktop:-/ga_find_wcetS time ./bubblesort < descending_i	ntegers.dat
real	6m0.275s	
user	0n0.275s	
sys	0m0.000s	
bagba-	-desktop:-/ga_find_wcet\$ time ./bubblesort < worstcaseinp	ut_GA_inputfile.dat
real	6m8.386s	
user	0m0.308s	
sys	6n0.000s	
bagba-	i-desktop:-/ga_find_wcet\$	

#### Genetic Algorithms. End-to-End Measurements.

## Single-core. Find Estimate of Worst-Case Execution Time.

Find resource characteristics of each thread/process	
Find timing requirements of each thread/process	Make sure that all threads/processes satisfy timing requirements when executing together on a shared computer platform
Find invocation pattern of	

#### Worst-Case Execution Time Analysis

File Edit View Search Terminal Help bagba-desktop:-/ga_find_wcetS more myconf 6 262144	
bagba-desktop:-/ga_find_wcetS more myconf 0 262144	
262144	
bagba-desktop:-/gw_find_wcet\$ ./ga_find_wcet myconf ./bubblesort 8.366359	
<pre>bagba-desktop:-/ga_find_wcet\$ time ./bubblesort &lt; ascending_integers.dat</pre>	
real 0n0.002s	
user 0m0.001s	
sys 0m0.001s	
<pre>bagba-desktop:-/ga_find_wcet5 time ./bubblesort &lt; descending_integers.dat</pre>	
real Om0.275s	
user 0n0.275s	
sys 0n0.000s	
bagba-desktop:-/ga_find_wcet\$ time :/bubblesort < worstcaseinput_GA_inputfile	ldat
real dm0.306s	
user 0m0,308s	
sys @n0.0005	
ba@ba-desktop:-/ga_find_wcet\$	

#### https://www.andrew.cmu.edu/user/banderss/s oftware/ga\_find\_wcet/ga\_find\_wcet.c

# Single-core. Response-Time Analysis.



Delay from arrival of thread i until it finishes = time for its own execution

time for higher-priority threads' execution

Compute upper bound on this.

╋

## Multicore. Find Estimate of Worst-Case Slowdown



Tool

						-	riated in the	and the second se			
		-	 -		-		1		there which a	Z	
1941 ( September 1 September 1	181100	11			1.00011	4	RC1.13.14(74000717471	(16.12. Lantacaberra)		44040	
lan i Ispini i Ispini i	1.		 111	•	1.000000		aC. 111100000011001		Same	anor	
Task I Indonesia Indonesia	10.000	LI.		£		44	gi ig san a san ta	i ja iga Damasana		<b>—</b>	
lan a Ingenet i Ingenet i		H				41.0	105.13 (475-000-00	CIT IS AN INCOME	hildensel	*****	
Last 2 Ingenerati 2 Ingenerati 2 Ingenerati 2	1.11	11	 ¥	-	1.04.0171	114	101, 11 1 10000000000000000000000000000	L (L) L (Lasertreinstein) (L) L (Lasertreinstein)	(Salation) (Salation)	27000 78000	
Taxi 3 Ingrant 1 Ingrant 2										1	
Tax. 1 Ingeneral											
-											
140.0											

https://www.andrew.cmu.edu/user/banderss/softwar e/pyschedanalysiscorunner\_including\_ga\_based\_pa rameter\_extraction/pyschedanalysiscorunner\_includi ng\_ga\_based\_parameter\_extraction.py

### Multicore. Find Estimate of Worst-Case Slowdown



#### Tool



Finds slowdown for each thread i, for each corunner set of threads co.

# Multicore. Compute upper bounds on response times



Tool



https://www.andrew.cmu.edu/user/banderss/softwar e/pyschedanalysiscorunner\_including\_ga\_based\_pa rameter\_extraction/pyschedanalysiscorunner\_includi ng\_ga\_based\_parameter\_extraction.py

### Multicore. Profiling



#### Tool

Configuration	Metric & Event Mapping	Metric Reports Activ	Partitions Heartheat	
				kimatee Roard(s)
	Sourd D Cere	Metric Mapping (HW / SW) Metric Name	Frequency Bake Tax	
- 0	Pa0600 0	App Care 8	- 190 - Official Of -	Mebles Bart Enabled & Red To
	PAREO 201 1 2	App Case 4	- 134 - CRassed ()	Makten Biop All
	P40500 2 2	App Cove 2	E 100 E Castie ()	Methics Exable All
	P40600 3 5	App Care 3	E TH E BRANN O	Matrices Dear Take
	PA0800 201 4 201	App Care A	- 100 - Canada ()	
	P40800 - 5 -	App Care 3	- 100 - Exame ()	
	P4080 0 6 0	App Care 6	100 E Canton ()	
		Save Metric/Tvent Mappings	1	
		Event Mapping		
the second	atd ID Core Parkton	Event Name	Frangements Bistists Tel	Contraction Contraction
2000		Lives 1283 Reader 198/100	I III Distance (	Events Bart Crabber & Nut To
		Linear 1283 Reader 198/100	199 - Distance C	Constant Hard Constant of Must To Farming Mage Ant
		Linear 128k Reader 500 Linear 128k Reader 500 Linear 128k Writer 100/108	Teathorn Calls In	Teacher Al
	Art (D)         Core         Parkkas           00007         0         1         1         1           00000         0         -         1         -         1           00000         -         0         -         1         -         1           00000         -         0         -         51         -         1         -           00000         -         0         -         51         -         51         -         1	Linear 128 Reader 100100 Linear 128 Reader 100 Linear 138 Write 100168 Linear 128 Write 100168	Teacharter	Commission     Commission
	Art (D)         Come         Parkkas           00007         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -         -	Linear 1288 Reader 191100 Linear 1288 Reader 191100 Linear 1288 Writer 190108 Linear 1288 Writer 198 -2889	Teachange Contract of Contract	Constant Bart Crasted & Wei To Bart Crasted & Wei To Craste Date Al Events Dear Tabe
	Art (D)         Come         Parklass           00000         =         0         5         1         1           00000         =         0         =         1         1         1           00000         =         0         =         1         1         1         1           00000         =         0         =         1         1         1         1           00000         =         0         =         2         1         1         1           00000         =         0         =         2         1         1         1           00000         =         0         =         2         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1         1 </td <td>Lonar 128 Rode 193100 Lonar 128 Rode 193100 Lonar 128 Wite 10018 Lonar 128 Wite 10018 Lonar 128 Wite 10018 Lonar 128 Wite 100</td> <td>Teatrony Control of Co</td> <td>Events Barr Crastion &amp; Nue Tr Control Duality AT Duality AT Duality AT</td>	Lonar 128 Rode 193100 Lonar 128 Rode 193100 Lonar 128 Wite 10018 Lonar 128 Wite 10018 Lonar 128 Wite 10018 Lonar 128 Wite 100	Teatrony Control of Co	Events Barr Crastion & Nue Tr Control Duality AT Duality AT Duality AT
	Art (D)         Correr         Parklass           00000         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0         0	Linear 128k Reader 198100 Linear 128k Reader 198100 Linear 128k Weller 100/188 Linear 128k Weller 100/188 Linear 128k Weller 100/188 -2890	Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Teacher Tea	Control Bart Exaction & Nucl To Exactly Evalue Ad Evalue Ad Evalue Ad Evalue Ad

#### INTERFERENCES DUE TO CONCURRENT ACCESSES TO SHARED RESOURCES EXAMPLE

.

Frontside L3 Gache

a500mc Core

32-kbyte

I-Cache

CoreNet<sup>\*M</sup> Coherency Fabric

PAMU

32

D-Cache

PAMU



- Core  $\rightarrow$  L1  $\rightarrow$  L2  $\rightarrow$  Interconnect  $\rightarrow$  DDR
- DDR  $\rightarrow$  Interconnect  $\rightarrow$  L2  $\rightarrow$  L1  $\rightarrow$  Core

PAMU

PAMU



DDR2/DDR3

amory Controll

Access Mont Line

PAMU

Memory Thrasher 256K - Distributions

Carnegie Mellon University Software Engineering Institute



Carnegie Mellon University Software Engineering Institute

#### **EXAMPLE – NXP T2080**



#### **AVMC MCP INTERFERENCE ANALYSIS**

#### MCP Interference Matrix

Resource	Issue	Analysis	Mitigation	Verification
Local Cache - Local cache miss - Local cache eviction - Cache coherency		<ul> <li>Performance metrics collection</li> <li>Cache contents inspection</li> <li>Use of cache simulators</li> </ul>	<ul> <li>Memory allocation adjustment</li> <li>Multiple memory controllers</li> <li>Cache inhibition</li> <li>Cache locking</li> </ul>	<ul> <li>Performance metrics collection</li> <li>Cache contents inspection</li> </ul>
Shared Cache	- Cache miss - Cache eviction - Cache coherency	<ul> <li>Performance metrics collection</li> <li>Analysis of configuration settings</li> <li>Use of cache simulators</li> </ul>	<ul> <li>Memory allocation adjustment</li> <li>Multiple memory controllers</li> <li>Cache inhibition</li> <li>Cache locking</li> <li>Cache coloring</li> <li>Cache partitioning</li> <li>Disabling shared cache</li> </ul>	<ul> <li>Off-core performance metrics collection</li> <li>Analysis of configuration settings</li> </ul>
Main Memory	- Access Latency	- Execution timing - Off-core Performance metrics collection	- Memory bank partitioning	<ul> <li>Execution timing</li> <li>Off-core Performance metrics collection</li> </ul>

Schedulability Analysis for Real-Time Multicore Software © 2023 Carnegie Mellon University

#### 1024-kbyte 64-bit RAM DDR2/DDR3 Frontside QorlQ P4080 Memory Controller L3 Cache Power Architecture<sup>TM</sup> e500mc Core 128-kbyte Backside 1024-kbyte Frontside 64-bit DDR2/DDR3 32-kbyte 2 Cache 32 D-Cache I-Cache L3 Cache Memory Controller eOpenPIC PreBoot **CoreNet™** Loader Coherency Fabric Security Peripheral Access Momt Unit Monitor PAMU PAMU PAMU PAMU PAMU Internal BootROM Power Mgmt Frame Manager Frame Manager Real Time Debug RapidIO SD/MMC Parse, Classify, Security Queue Parse, Classify, Message Watchpoint 2x DMA eLBC Unit Mgr Distribute Distribute 4.0 Cross SPI (RMU) Trigger 2x DUART Buffer Buffer 4x 12C CoreNet Perf Pattern Test Match Buffer 1GE 1GE 1GE 1GE Monitor Trace Port/ PCIe PCIe 2x USB 2.0/ULPI sRIO Mgr 10GE 10GE Engine 1.0 . 100 1000 sRIO PCIe 2.0 1GE 1GE 1GE 1GE Aurora Clocks/Reset GPIO CCSR 18-Lane 5-GHz SERDES

#### NXP P4080 SYSTEM ON A CHIP (SOC)

QorlQ<sup>™</sup> P4080 Communications Processor Product Brief, Rev. 1

#### Freescale Semiconductor

SOURCE: https://www.nxp.com/products/processors-and-microcontrollers/power-architecture-processors/qoriq-platforms/p-series/qoriq-p4080-p4040-p4081-multicorecommunications-processors:P4080

#### MCP Multi-Core Analysis Framework

CoreNet Coherency Fabric





64-bit DDR2/DDR3 lemory Control

64-bit DDR2/DDR3 Memory Control

### Conclusion

The two-step framework for timing verification has been known for a long time in the academic research literature on real-time systems.

We have presented it to the avionics community.

There are tools that support activities of this two-step framework. Some from SEI and AvMC presented here.

Some tools can be downloaded here: https://www.andrew.cmu.edu/user/banderss/projects.html

### Thanks!