REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
27-01-2023	Final Report	1-Sep-2011 - 31-Aug-2014

4. TITLE AND SUBTITLE
Final Report: Compositional Framework for Complex Real-time Systems on Multicore Platforms

5a. CONTRACT NUMBER
W911NF-11-1-0403

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Insup Lee, Linh Thi Xuan Phan, Oleg Sokolsky

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
University of Pennsylvania
Office of Research Services
3451 Walnut Street, 5th Floor
Philadelphia, PA 19104 -6205

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
59800-NC.13

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
This project develops advanced theoretical foundations and systems design techniques for real-time virtualization and cloud computing platforms to support safety- and mission-critical systems. Our approach tightly integrates and fundamentally advances recent real-time compositional scheduling research and virtualization technologies to achieve provable performance guarantees while optimizing resources.

Concretely, the major goals of the project include the development of (i) cache-aware compositional scheduling

15. SUBJECT TERMS
compositional analysis, timing guarantees, virtualization, multicore

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Insup Lee
UU	UU	UU	UU		19b. TELEPHONE NUMBER 215-898-3532

RPPR Final Report
as of 27-Jan-2023

Agency Code: 21XD

Proposal Number: 59800NC **Agreement Number: W911NF-11-1-0403**
INVESTIGATOR(S):

Name: Insup Lee lee@cis.up
Email: lee@cis.upenn.edu
Phone Number: 2158983532
Principal: Y

Name: Oleg Sokolsky Phd
Email: sokolsky@cis.upenn.edu
Phone Number: 2158984448
Principal: N

Organization: **University of Pennsylvania**
Address: Office of Research Services, Philadelphia, PA 191046205
Country: USA
DUNS Number: 042250712 EIN: 231352685

Report Date: 30-Nov-2014 Date Received: 27-Jan-2023
Final Report for Period Beginning 01-Sep-2011 and Ending 31-Aug-2014
Title: Compositional Framework for Complex Real-time Systems on Multicore Platforms
Begin Performance Period: 01-Sep-2011 **End Performance Period:** 31-Aug-2014
Report Term: 0-Other

Submitted By: Insup Lee Email: lee@cis.upenn.edu
Phone: (215) 898-3532

STEM Degrees: **STEM Participants:**

Major Goals: Modern real-time systems are becoming increasingly complex, autonomous, and resource-constrained. A modern vehicular control system, for instance, has as many as 100 microprocessors running simultaneously and controlling thousands of software functions. Further, to enhance flexibility and to reduce cost, weight and power, these systems are increasingly being deployed on shared multicore platforms, instead of isolated physical hosts. Due to these trends, existing analysis techniques – which were developed for much simpler systems – are becoming inadequate, as they cannot scale to this unprecedented level of complexity.

To address the above challenge, this project aimed to develop a novel compositional scheduling and analysis framework for safety-critical and mission-critical real-time systems on multicore architectures. Through new resource-aware interfaces for components and interface analysis methods, the framework not only enables efficient and accurate compositional modeling and analysis of large-scale open systems with tight resource constraints, but it also provides a foundation for component-based development with timing guarantees. Besides these theoretical innovations, the project developed tools and concrete system implementations to demonstrate the utility and effectiveness of the proposed framework, as well as to facilitate technology transfer and follow-on research on robust and interoperable real-time systems.

Accomplishments: 1. Real-time multicore virtualization platform

Virtualization provides a convenient means for realizing component-based development, as it ensures functional isolation among applications running in different virtual machines; however, existing virtualization platforms – such as Xen and VMWare – do not provide temporal isolation, which is crucial for real-time systems. At the same time, multicore processors are becoming increasingly common in modern real-time systems, due to their benefits in computational power and cost reduction. The combination of virtualization and multicore makes it especially challenging to achieve timing guarantees and performance isolation among components. In particular, the overhead due to cache misses a component experiences is difficult to quantify, as it depends not only on the interference among its tasks but also on the interference between the virtual processors and tasks within other components.

To enable real-time support for component-based multicore virtualization, we have developed several extensions of the existing compositional analysis framework that address the cache-interference challenges on a multicore and virtualization platform. Specifically, we have investigated (i) new resource-aware interfaces that enable resource-efficient scheduling of the virtual machines on a multicore platform, (ii) cache-related overhead accounting techniques that consider virtualization- and multicore-specific characteristics, such as the various types of events that cause cache misses in the presence of virtualization, as well as the mapping between virtual resources and physical resources; and (iii) cache-aware multicore compositional analysis methods.

First, we had designed a new cache-aware compositional analysis technique that takes into account the cache overhead on multicore virtualization platforms. For this, we developed two approaches for overhead accounting: TASK-CENTRIC, which inflates the worst-case execution time of each task with the cache overhead each task experiences; and MODEL-CENTRIC, which uses the effective resource supply of the physical platform after considering the combined overhead of all tasks. We then developed a new cache-aware interface computation method based on these accounting approaches. Our evaluation on synthetic task sets shows that our technique can safely account for the cache overhead experienced by the system, while achieving significant resource savings compared to a baseline method. This result was presented at RTSS 2013.

We extended the above result to reduce the cache-related overhead and to improve the accuracy and efficiency of the analysis. Specifically, we have developed the following enhancements:

- We identified situations when the existing supply bound function (SBF) for the Multiprocessor Periodic Resource (MPR) model proposed earlier is conservative. We then introduced a tighter SBF for the MPR model to achieve more resource-efficient interfaces, and proved the correctness of our new SBF computation;
- We developed a new task-centric overhead accounting method that incorporates an upper bound on the number of VCPUs of an interface; this helps to eliminate the overhead caused by VCPU-preemption and VCPU-completion events. In addition, we established the relationship among all three cache-aware analysis methods using several examples and a formal proof.
- We extended the cache-aware interface analysis method to consider different cache overhead values for different tasks (instead of using only the maximum for all tasks) in the system. This extension helps tighten the analysis and improve the resource use for tasks with diverse cache behaviors.

We conducted a comprehensive set of experiments to evaluate the different analysis approaches under various settings. The evaluation results showed that the proposed extension is effective in both optimizing resource use and enhancing scalability. The extended framework has been published in the Real-Time Systems Journal, 2015.

2. Scheduling of multi-mode mixed-criticality real-time systems on multicore platforms

Due to the increase in autonomy and adaptivity requirements, modern real-time systems often consist of multiple modes of operation, where each mode represents a system configuration and each mode transition represents a reconfiguration in response to changes in the system or environment. At the same time, these systems are also becoming increasingly mixed-critical: they typically contain a mixture of tasks with different degrees of safety criticality. The combination of multi-mode and mixed-criticality makes scheduling highly challenging, especially because both the active tasks and their criticality can change as the system changes its mode.

In collaboration with CMU, we have developed a partitioned multicore scheduling technique for multi-mode real-time systems with mixed-criticality constraints. Our technique includes a partitioning algorithm for packing tasks to cores and a mixed-criticality scheduling algorithm for scheduling tasks on each core, both of which consider the impact of criticality changes during mode changes. The developed scheduler can guarantee both timing performance and safety-criticality requirements of the systems under dynamic operating conditions and system reconfigurations, and it can save up to half the resource compared to a mode-agnostic scheme. We have also developed an opern-source implementation of the scheduler in the Linux operating system. The results have been presented at RTAS 2014.

3. Architecture support for compositional scheduling

We had developed RT-Xen 1.0, a real-time virtualization platform based on Xen that realizes our theoretical compositional scheduling framework on single-core platforms. As multicore processors are becoming common in modern real-time systems, we have further developed a multicore extension RT-Xen 2.0, containing a new suite of multicore real-time VM scheduling policies. It provides a high degree of flexibility in terms of scheduling configurations: it allows combinations of global and partitioned scheduling strategies, static and dynamic priority schemes, periodic and deferrable servers. It also supports multiple resource interfaces for VMs that are compatible

with the multicore compositional schedulability analysis methods, thus enabling designers to directly use these theoretical results. We have conducted a comprehensive experimental evaluation of the RT-Xen 2.0 platform using synthetic workloads. Our evaluation demonstrates that the platform can provide timing guarantees to components running within the VMs in practice, and can be realized within Xen at moderate overhead. These results have been presented at EMSOFT 2014.

4. Cache partitioning techniques

On a multicore platform, the cache interference among tasks (components) can increase the response times of the tasks (components) substantially, resulting in higher resource needs. In particular, the interference due to concurrent cache accesses may evict the useful cache blocks of each other from the cache, resulting in cache misses.

We have investigated cache partition techniques for compositional real-time systems to reduce the potential cache interference among components and among tasks within a component. We have explored a range of partitioning approaches – including dynamic, static, and hybrid – using both hardware and software-based mechanisms. In addition, we have also been exploring new cache-aware scheduling techniques to co-schedule CPU and cache resources to minimize cache interference while improving the utilization of CPU and cache resources concurrently. We achieved promising initial results on cache-aware scheduling algorithms for multicore platforms.

5. Tool development

We have enhanced the CARTS toolset, our platform-independent analysis tool, with the new theoretical results on compositional multicore scheduling and analysis.

Training Opportunities: Meng Xu, a Ph.D. student, has been working on the development of RT-Xen platform. He has had an opportunity to study the inner working of the popular Xen platform and deeply engage with the Xen open-source development community.

RPPR Final Report
as of 27-Jan-2023

Results Dissemination: A number of publications have been presented at the top venues in the real-time systems community:

[1] Realizing Compositional Scheduling through Virtualization. J. Lee, S. Xi, S. Chen, L. T. X. Phan, C. Gill, I. Lee, C. Lu and O. Sokolsky. 18th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), Beijing, China, Apr. 2012

[2] An Empirical Analysis of Scheduling Techniques for Real-time Cloud-based Data Processing. L.T.X. Phan, Z. Zhang, Qi Zheng, B.T. Loo, and I. Lee. 4th IEEE International Workshop on Real-time Service-Oriented Architectures and Applications (RTSOAA), Irvine, California, Dec 2011.

[3] Towards A Compositional Multi-Modal Framework for Adaptive Cyber-Physical Systems. L.T.X. Phan and I. Lee. 1st International Workshop on Cyber-Physical Systems, Networks, and Applications (CPSNA), Toyama, Japan, Aug 2011.

[4] Cache-Aware Compositional Analysis of Real-Time Multicore Virtualization Platforms.
Meng Xu, Linh T. X. Phan, Insup Lee, Oleg Sokolsky, Sisu Xi, Chenyang Lu, and Christopher Gill. In Proceedings of the IEEE Real Time Systems Symposium (RTSS), Vancouver, Canada, Dec. 2013.

[5] Partitioned Scheduling of Multi-Modal Mixed-Criticality Real-Time Systems on Multiprocessor Platforms. Dionisio de Niz and Linh T. X. Phan. In Proceedings of the 20th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), Berlin, Germany, Apr. 2014.

[6]. Cache-Aware Compositional Analysis of Real-Time Multicore Virtualization Platforms.
Meng Xu, Linh T. X. Phan, Oleg Sokolsky, Sisu Xi, Chenyang Lu, Christopher Gill and Insup Lee. Real-Time Systems Journal, April 2015.

[7] Real-Time Multi-Core Virtual Machine Scheduling in Xen. Sisu Xi, Meng Xu, Chenyang Lu, Linh T. X. Phan, Chris Gill, Oleg Sokolsky, and Insup Lee. Proceedings of the ACM International Conference on Embedded Software (EMSOFT), New Delhi, India, Oct. 2014.

[8] Fluid Model-based Mixed-Criticality Scheduling on Multiprocessors, Jaewoo Lee, Kieu-My Phan, Xiaozhe Gu, Jiyeon Lee, Arvind Easwaran, Insik Shin and Insup Lee, IEEE RTSS, Dec 2014. (Best Paper runner-up)

In addition, we have developed tutorials on compositional scheduling, delivered at the Embedded Systems Week, held in Japan in Aug 2011 and at the Cyber-Physical Systems Week in April, 2012.

Honors and Awards: - The paper “Cache-Aware Compositional Analysis of Real-Time Multicore Virtualization Platforms” was nominated for the Best Paper Award at the RTSS 2013 conference.

- The paper “Fluid Model-based Mixed-Criticality Scheduling on Multiprocessors” was nominated for the Best Paper Award at the IEEE RTSS 2014 conferences.

- Co-PI Oleg Sokolsky was promoted from Associate Research Professor to Full Research Professor.

- Dr. Linh Thi Xuan Phan was promoted from Associate Research Professor to tenure-track Assistant Professor

Protocol Activity Status:

RPPR Final Report
as of 27-Jan-2023

Technology Transfer: The CARTS analysis toolset and the RT-Xen real-time virtualization platform are distributed as open source, thus making the research results supported by this project available to a wide range of system developers and organizations. The source code is publicly available via the following websites:

CARTS: http://rtg.cis.upenn.edu/carts/
RT-Xen: https://sites.google.com/site/realtimexen/

The RT-Xen prototype was also linked to the Xen developer website (http://blog.xen.org/index.php/2013/11/27/rt-xen-real-time-virtualization-in-xen/). At the time of this report, RT-Xen has been included as part of the official Xen release, enabling many cloud users to benefits from our research results.

We have also been exploring opportunities for transitioning the developed technology into advanced development in industry sectors. In particular, we are in communication with colleagues at General Motors on adapting the compositional scheduling framework to provide timing guarantee and safety assurance for future autonomous vehicles.

PARTICIPANTS:

Participant Type: PD/PI
Participant: Insup Lee
Person Months Worked: 4.00 **Funding Support:**
Project Contribution:
National Academy Member: N

Participant Type: Co PD/PI
Participant: Oleg Sokolsky
Person Months Worked: 8.00 **Funding Support:**
Project Contribution:
National Academy Member: N

Participant Type: Faculty
Participant: Linh Phan
Person Months Worked: 4.00 **Funding Support:**
Project Contribution:
National Academy Member: N

Participant Type: Graduate Student (research assistant)
Participant: Meng Xu
Person Months Worked: 15.00 **Funding Support:**
Project Contribution:
National Academy Member: N

Participant Type: Graduate Student (research assistant)
Participant: Jaewoo Lee
Person Months Worked: 12.00 **Funding Support:**
Project Contribution:
National Academy Member: N

RPPR Final Report
as of 27-Jan-2023

ARTICLES:

Publication Type: Journal Article Peer Reviewed: Y **Publication Status:** 1-Published
Journal: Real-Time Systems
Publication Identifier Type: DOI Publication Identifier: 10.1007/s11241-012-9151-3
Volume: 48 Issue: 4 First Page #: 0
Date Submitted: Date Published:
Publication Location:
Article Title: State-based scheduling with tree schedules: analysis and evaluation
Authors:
Keywords: schedulability analysis, real-time systems, distributed systems
Abstract: Distributed real-time systems require bounded communication delays and achieve them by means of a predictable and verifiable control mechanism for the communication medium. Real-time bus arbitration mechanisms control access to the medium and guarantee bounded communication delays. These arbitration mechanisms can be static dispatch tables or dynamic, algorithmic approaches. In this work, we introduce a real-time bus arbitration mechanism called tree schedules that takes the best parts of both sides: It can be analyzed like static dispatch tables, and it provides a certain degree of flexibility similar to algorithmic approaches. We present tree schedules as a framework to specify real-time traffic and introduce mechanisms to analyze it. We discuss how tree schedules can capture application-specific behavior in a time-triggered state-based supply model by means of conditional branching built into the model. We present analysis results for this model specifically aiming at schedulability
Distribution Statement: 1-Approved for public release; distribution is unlimited.
Acknowledged Federal Support:

Publication Type: Journal Article Peer Reviewed: Y **Publication Status:** 1-Published
Journal: ACM Transactions on Cyber-Physical Systems
Publication Identifier Type: Publication Identifier:
Volume: 1 Issue: 1 First Page #: 1
Date Submitted: 6/18/18 12:00AM Date Published: 5/1/18 4:00AM
Publication Location:
Article Title: Real-Time Middleware for Cyber-Physical Event Processing
Authors: C. Wang, C. Gill, C. Lu
Keywords: middleware, real-time, CPS
Abstract: Cyber-physical systems (CPS) involve tight integration of cyber (computation) and physical domains, and both the effectiveness and correctness of a CPS application may rely on successful enforcement of constraints such as bounded latency and temporal validity subject to physical conditions. For many such systems (e.g., edge computing in the Industrial Internet of Things), it is desirable to enforce such constraints within a common middleware service (e.g., during event processing). In this article, we introduce CPEP, a new real-time middleware for cyber-physical event processing, with (1) extensible support for complex event processing operations, (2) execution prioritization and sharing, (3) enforcement of time consistency with load shedding, and (4) efficient memory management and concurrent data processing. We present the design, implementation, and empirical evaluation of CPEP and show that it can (1) support complex operations needed by many applications, (2) schedule data process
Distribution Statement: 3-Distribution authorized to U.S. Government Agencies and their contractors
Acknowledged Federal Support: **Y**

Publication Type: Journal Article Peer Reviewed: Y **Publication Status:** 1-Published
Journal: IEEE Transactions on Parallel and Distributed Systems
Publication Identifier Type: DOI Publication Identifier: 10.1109/TPDS.2017.2777454
Volume: 29 Issue: 4 First Page #: 789
Date Submitted: 6/18/18 12:00AM Date Published: 4/1/18 12:00AM
Publication Location:
Article Title: Blocking Analysis for Spin Locks in Real-Time Parallel Tasks
Authors: Son Dinh, Jing Li, Kunal Agrawal, Chris Gill, Chenyang Lu
Keywords: Real-time systems, Schedules, Synchronization, Computational modeling, Numerical models, Processor scheduling
Abstract: In recent years, there has been significant interest in developing real-time schedulers for parallel tasks. Most of that research has concentrated on idealized task models where tasks do not access any shared resources protected with locks. In this paper, we consider the problem of scheduling parallel tasks which experience contention due to shared resources. In particular, we provide a schedulability test for federated scheduling by deriving blocking time analyses for parallel tasks that access shared resources protected by FIFO-ordered and priority-ordered spin locks. Our numerical evaluation on randomly generated task sets indicates that priority-ordered locks generally provide better schedulability results than FIFO-ordered locks. We also incorporated both FIFO-ordered and priority-ordered spin lock implementations into a federated scheduling platform. Via empirical evaluations, we found that priority-ordered locks also have better performance than FIFO-ordered locks in practice.
Distribution Statement: 3-Distribution authorized to U.S. Government Agencies and their contractors
Acknowledged Federal Support: **Y**

Publication Type: Journal Article Peer Reviewed: Y **Publication Status:** 1-Published
Journal: Real-Time Systems Journal
Publication Identifier Type: DOI Publication Identifier: https://doi.org/10.1007/s11241-015-9223-2
Volume: 51 Issue: First Page #: 675
Date Submitted: 1/24/23 12:00AM Date Published: 4/29/15 1:51AM
Publication Location:
Article Title: Cache-aware compositional analysis of real-time multicore virtualization platforms
Authors: Meng Xu, Linh Thi Xuan Phan, Oleg Sokolsky, Sisu Xi, Chenyang Lu, Christopher Gill, Insup Lee
Keywords: Compositional analysis, Interface, Cache-aware, Multicore, Virtualization
Abstract: Multicore processors are becoming ubiquitous, and it is becoming increasingly common to run multiple real-time systems on a shared multicore platform. While this trend helps to increase performance, it also makes it more challenging to achieve timing guarantees and functional isolation. We achieve functional isolation via virtualization. However, virtualization also introduces many challenges to the multicore timing analysis; e.g., the overhead due to cache misses becomes harder to predict due to additional sources of interference. We present a cache-aware compositional analysis technique that can be used to ensure timing guarantees of components scheduled on a multicore virtualization platform. Our technique accounts for the cache-related overhead in the components' interfaces and addresses the new virtualization-specific challenges in the overhead analysis. To demonstrate the utility of our technique, we perform an extensive evaluation on randomly generated workloads.
Distribution Statement: 2-Distribution Limited to U.S. Government agencies only; report contains proprietary info
Acknowledged Federal Support: **Y**

CONFERENCE PAPERS:

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time and Embedded Technology and Applications Symposium
Date Received: 24-Jan-2023 Conference Date: 17-Apr-2012 Date Published:
Conference Location: Beijing, China
Paper Title: Realizing Compositional Scheduling through Virtualization
Authors: Jaewoo Lee, Sisu Xi, Sanjian Chen, Linh T.X. Phan, Chris Gill, Insup Lee, Chenyang Lu, Oleg Sokolsky
Acknowledged Federal Support: **Y**

RPPR Final Report
as of 27-Jan-2023

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE International Workshop on Real-time Service-Oriented Architectures and Applications
Date Received: 26-Jan-2023 Conference Date: 14-Dec-2011 Date Published:
Conference Location: Irvine, CA
Paper Title: An Empirical Analysis of Scheduling Techniques for Real-time Cloud-based Data Processing
Authors: Linh T. X. Phan, Zhuoyao Zhang, Qi Zheng, Boon Thau Loo, Insup Lee
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time and Embedded Technology and Applications Symposium
Date Received: 24-Jan-2023 Conference Date: 09-Apr-2013 Date Published:
Conference Location: Philadelphia, PA
Paper Title: Overhead-Aware Compositional Analysis of Real-Time Systems.
Authors: Linh T.X. Phan, Meng Xu, Jaewoo Lee, Insup Lee, and Oleg Sokolsky
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time Systems Symposium
Date Received: 26-Jan-2023 Conference Date: 04-Dec-2013 Date Published:
Conference Location: Vancouver, Canada
Paper Title: Cache-AwareCompositional Analysis of Real-Time Multicore Virtualization Platforms
Authors: Meng Xu, Linh T. X. Phan, Insup Lee, Oleg Sokolsky, Sisu Xi, Chenyang Lu, and Christopher Gill
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE International Symposium on Rapid System Prototyping
Date Received: 26-Jan-2023 Conference Date: 03-Oct-2012 Date Published:
Conference Location: Tampere, Finland
Paper Title: A Model-Based I/O Interface Synthesis Framework for the Cross-Platform Software Model.
Authors: BaekGyu Kim, Linh T.X. Phan, Insup Lee, Oleg Sokolsky
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)
Date Received: 20-Oct-2016 Conference Date: 11-Apr-2016 Date Published: 11-Apr-2016
Conference Location: Vienna, Austria
Paper Title: Analysis and Implementation of Global Preemptive Fixed-Priority Scheduling with Dynamic Cache Allocation
Authors: Xu, Meng; Phan, Linh Thi Xuan; Choi, H.-Y; Lee, Insup
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)
Date Received: 20-Oct-2016 Conference Date: 11-Apr-2016 Date Published: 11-Apr-2016
Conference Location: Vienna, Austria
Paper Title: Analysis and Implementation of Global Preemptive Fixed-Priority Scheduling with Dynamic Cache Allocation
Authors: Xu, Meng; Phan, Linh Thi Xuan; Choi, H.-Y; Lee, Insup
Acknowledged Federal Support: **Y**

RPPR Final Report
as of 27-Jan-2023

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)
Date Received: 18-Jun-2018 Conference Date: 11-Apr-2018 Date Published: 11-Apr-2018
Conference Location: Porto, Portugal
Paper Title: Multi-Mode Virtualization for Soft Real-Time Systems
Authors: • H. Li, M. Xu, C. Li, C. Lu, C. Gill, L.T.X. Phan, I. Lee, O. Sokolsky
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)
Date Received: 18-Jun-2018 Conference Date: 11-Apr-2018 Date Published: 11-Apr-2018
Conference Location: Porto, Portugal
Paper Title: SafeMC: A system for the design and evaluation of mode change protocols
Authors: Tianyang Chen, Linh Thi Xuan Phan
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 4-Under Review
Conference Name: IEEE Real-Time Systems Symposium (RTSS)
Date Received: 18-Jun-2018 Conference Date: 11-Dec-2018 Date Published: 12-Dec-2018
Conference Location: Nashville, Tennessee
Paper Title: A Partial VCPU Is Enough
Authors: • H. Li, M. Xu, C. Li, C. Lu, C. Gill, L.T.X. Phan, I. Lee, O. Sokolsky
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 4-Under Review
Conference Name: IEEE Real-Time Systems Symposium (RTSS)
Date Received: 18-Jun-2018 Conference Date: 11-Dec-2018 Date Published: 12-Dec-2018
Conference Location: Nashville, Tennessee
Paper Title: Holistic resource allocation for multicore real-time systems
Authors: Meng Xu, Linh Thi Xuan Phan, Hyon-Young Choi, Yuhan Lin, Haoran Li, Chenyang Lu, Insup Lee.
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: 14th International Conference on Embedded Software (EMSOFT)
Date Received: 24-Jan-2023 Conference Date: 13-Oct-2014 Date Published:
Conference Location: New Delhi, India
Paper Title: Real-Time Multi-Core Virtual Machine Scheduling in Xen
Authors: Sisu Xi, Meng Xu, Chenyang Lu, Linh T.X. Phan, Christopher Gill, Oleg Sokolsky, and Insup Lee
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: 2014 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)
Date Received: 26-Jan-2023 Conference Date: 15-Apr-2014 Date Published:
Conference Location: Berlin, Germany
Paper Title: Partitioned scheduling of multi-modal mixed-criticality real-time systems on multiprocessor platforms
Authors: Dionisio de Niz, Linh T.X. Phan
Acknowledged Federal Support: **Y**

RPPR Final Report
as of 27-Jan-2023

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: 2014 IEEE Real-Time Systems Symposium (RTSS)
Date Received: 26-Jan-2023 Conference Date: 02-Dec-2014 Date Published:
Conference Location: Rome, Italy
Paper Title: MC-Fluid: Fluid Model-Based Mixed-Criticality Scheduling on Multiprocessors
Authors: Jaewoo Lee, Kieu-My Phan, Xiaozhe Gu, Jiyeon Lee, Arvind Easwaran, Insik Shin, Insup Lee
Acknowledged Federal Support: **Y**

Publication Type: Conference Paper or Presentation **Publication Status:** 1-Published
Conference Name: 2011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)
Date Received: 26-Jan-2023 Conference Date: 28-Aug-2011 Date Published:
Conference Location: Toyama, Japan
Paper Title: Towards a Compositional Multi-modal Framework for Adaptive Cyber-physical Systems
Authors: Linh T.X. Phan, Insup Lee
Acknowledged Federal Support: **N**

DISSERTATIONS:

Publication Type: Thesis or Dissertation
Institution: University of Pennsylvania
Date Received: 26-Jan-2023 Completion Date: 5/18/18 9:50PM
Title: Real-Time Multicore Virtualization
Authors: Meng Xu
Acknowledged Federal Support: **N**

WEBSITES:

URL: https://sites.google.com/site/realtimexen/
Date Received: 18-Jun-2018
Title: RT-Xen: Real-Time Virtualization
Description: The official website for the RT-Xen software
URL: https://rtg.cis.upenn.edu/carts/
Date Received: 18-Jun-2018
Title: CARTS: Compositional Analysis of Real-Time Systems
Description: The official website for the CARTS toolset

Partners

,

RPPR Final Report

as of 27-Jan-2023

I certify that the information in the report is complete and accurate:
Signature: Insup Lee
Signature Date: 1/27/23 11:42AM

Abstract

Modern real-time systems are becoming increasingly complex, autonomous, and resource-constrained. A modern vehicular control system, for instance, has as many as 100 microprocessors running simultaneously and controlling thousands of software functions. Further, to enhance flexibility and to reduce cost, weight and power, these systems are increasingly being deployed on shared multicore platforms, instead of isolated physical hosts. Due to these trends, existing analysis techniques – which were developed for much simpler , mostly uniprocessor systems – are becoming inadequate, as they cannot scale to this unprecedented level of complexity. This project aimed at developing an approach for the design of component-based real-time systems and their guaranteed deployment on multi-core architectures. Goals of the project included both new theoretical advances in compositional scheduling and analysis, as well as open-source implementation of the development and deployment framework.

Objectives

To address the challenges of building real-time support for modern applications and computing platforms, this project had the following objectives:

(1) To develop a novel compositional scheduling and analysis framework for safety-critical and mission-critical real-time systems on multi-core architectures based on resource-aware interfaces for components and interface analysis methods.

(2) To develop cache management techniques to enable compositional management of tight resource constraints.

(3) To develop new scheduling algorithms for component-based mixed-criticality systems.

(4) To implement the framework and scheduling algorithms in an existing open-source virtualization platform.

(5) To implement developed analysis algorithms in an open source tool.

Accomplishments

1. Real-time multicore virtualization platform

Virtualization provides a convenient means for realizing component-based development, as it ensures functional isolation among applications running in different virtual machines; however, existing virtualization platforms – such as Xen and VMWare – do not provide temporal isolation, which is crucial for real-time systems. At the same time, multicore processors are becoming increasingly common in modern real-time systems, due to their benefits in computational power and cost reduction. The combination of virtualization and multicore makes it especially challenging to achieve timing guarantees and performance isolation among components. In particular, the overhead due to cache misses a component experiences is difficult to quantify, as it depends not only on the interference among its tasks but also on the interference between the virtual processors and tasks within other components.

To enable real-time support for component-based multicore virtualization, we have developed several extensions of the existing compositional analysis framework that address the cache-interference challenges on a multicore and virtualization platform. Specifically, we have investigated (i) new resource-aware interfaces that enable resource-efficient scheduling of the virtual machines on a multicore platform, (ii) cache-related overhead accounting techniques that consider virtualization- and multicore-specific characteristics, such as the various types of events that cause cache misses in the presence of virtualization, as well as the mapping between virtual resources and physical resources; and (iii) cache-aware multicore compositional analysis methods.

First, we had designed a new cache-aware compositional analysis technique that takes into account the cache overhead on multicore virtualization platforms. For this, we developed two approaches for overhead accounting: TASK-CENTRIC, which inflates the worst-case execution time of each task with the cache overhead each task experiences; and MODEL-CENTRIC, which uses the effective resource supply of the physical platform after considering the combined overhead of all tasks. We then developed a new cache-aware interface computation method based on these accounting approaches. Our evaluation on synthetic task sets shows that our technique can safely account for the cache overhead experienced by the system, while achieving significant resource savings compared to a baseline method. This result was presented at RTSS 2013.

We extended the above result to reduce the cache-related overhead and to improve the accuracy and efficiency of the analysis. Specifically, we have developed the following enhancements:

- We identified situations when the existing supply bound function (SBF) for the Multiprocessor Periodic Resource (MPR) model proposed earlier is conservative. We then introduced a tighter SBF for the MPR model to achieve more resource-efficient interfaces, and proved the correctness of our new SBF computation;
- We developed a new task-centric overhead accounting method that incorporates an upper bound on the number of VCPUs of an interface; this helps to eliminate the overhead caused by VCPU-preemption and VCPU-completion events. In addition, we established the relationship among all three cache-aware analysis methods using several examples and a formal proof.
- We extended the cache-aware interface analysis method to consider different cache overhead values for different tasks (instead of using only the maximum for all tasks) in the system. This extension helps tighten the analysis and improve the resource use for tasks with diverse cache behaviors.
- We conducted a comprehensive set of experiments to evaluate the different analysis approaches under various settings. The evaluation results showed that the proposed extension is effective in both optimizing resource use and enhancing scalability. The extended framework has been published in the Real-Time Systems Journal, 2015.

2. Scheduling of multi-mode mixed-criticality real-time systems on multicore platforms

Due to the increase in autonomy and adaptivity requirements, modern real-time systems often consist of multiple modes of operation, where each mode represents a system configuration and each mode transition represents a reconfiguration in response to changes in the system or environment. At the same time, these systems are also becoming increasingly mixed-critical: they typically contain a mixture of tasks with different degrees of safety criticality. The combination of multi-mode and mixed-criticality makes scheduling highly challenging, especially because both the active tasks and their criticality can change as the system changes its mode.

In collaboration with CMU, we have developed a partitioned multicore scheduling technique for multi-mode real-time systems with mixed-criticality constraints. Our technique includes a partitioning algorithm for packing tasks to cores and a mixed-criticality scheduling algorithm for scheduling tasks on each core, both of which consider the impact of criticality changes during mode changes. The developed scheduler can guarantee both timing performance and safety-criticality requirements of the systems under dynamic operating conditions and system reconfigurations, and it can save up to half the resource compared to a mode-agnostic scheme. We have also developed an opern-source implementation of the scheduler in the Linux operating system. The results have been presented at RTAS 2014.

3. Architecture support for compositional scheduling

We had developed RT-Xen 1.0, a real-time virtualization platform based on Xen that realizes our theoretical compositional scheduling framework on single-core platforms. As multicore processors are becoming common in modern real-time systems, we have further developed a multicore extension RT-Xen 2.0, containing a new suite of multicore real-time VM scheduling policies. It provides a high degree of flexibility in terms of scheduling configurations: it allows combinations of global and partitioned scheduling strategies, static and dynamic priority schemes, periodic and deferrable servers. It also supports multiple resource interfaces for VMs that are compatible with the multicore compositional schedulability analysis methods, thus enabling designers to directly use these theoretical results. We have conducted a comprehensive experimental evaluation of the RT-Xen 2.0 platform using synthetic workloads. Our evaluation demonstrates that the platform can provide timing guarantees to components running within the VMs in practice, and can be realized within Xen at moderate overhead. These results have been presented at EMSOFT 2014.

4. Cache partitioning techniques

On a multicore platform, the cache interference among tasks (components) can increase the response times of the tasks (components) substantially, resulting in higher resource needs. In particular, the interference due to concurrent cache accesses may evict the useful cache blocks of each other from the cache, resulting in cache misses.

We have investigated cache partition techniques for compositional real-time systems to reduce the potential cache interference among components and among tasks within a component. We have explored a range of partitioning approaches – including dynamic, static, and hybrid – using both hardware and software-based mechanisms. In addition, we have also been exploring new cache-aware scheduling techniques to co-schedule CPU and cache resources to minimize cache interference while improving the utilization of CPU and cache resources concurrently. We achieved promising initial results on cache-aware scheduling algorithms for multicore platforms.

5. Analysis tool development

CARTS is a platform-independent analysis tool that can be used to perform schedulability analysis for compositional real-time systems and automatically generate resource interfaces needed for the compositional analysis of real-time systems. We have enhanced the CARTS toolset, our platform-independent analysis tool, with the new theoretical results on compositional multicore scheduling and analysis described above.