

Mälardalen University Press Dissertations
No. 232

**REAL-TIME COMMUNICATION OVER
WORMHOLE-SWITCHED ON-CHIP NETWORKS**

Meng Liu

2017



School of Innovation, Design and Engineering

Copyright © Meng Liu, 2017

ISBN 978-91-7485-332-2

ISSN 1651-4238

Printed by E-Print AB, Stockholm, Sweden

Mälardalen University Press Dissertations
No. 232

REAL-TIME COMMUNICATION OVER WORMHOLE-SWITCHED ON-CHIP NETWORKS

Meng Liu

Akademisk avhandling

som för avläggande av i datavetenskap vid Akademin för innovation, design och teknik
kommer att offentligens försvaras tisdagen den 20 juni 2017, 09.15 i Gamma, Västerås.

Fakultetsopponent: Professor Christian Fraboul, University of Toulouse



Akademin för innovation, design och teknik

Abstract

In a modern industrial system, the requirement on computational capacity has increased dramatically, in order to support a higher number of functionalities, to process a larger amount of data or to make faster and safer run-time decisions. Instead of using a traditional single-core processor where threads can only be executed sequentially, multi-core and many-core processors are gaining more and more attentions nowadays. In a multi-core processor, software programs can be executed in parallel, which can thus boost the computational performance. Many-core processors are specialized multi-core processors with a larger number of cores which are designed to achieve a higher degree of parallel processing. An on-chip communication bus is a central intersection used for data-exchange between cores, memory and I/O in most multi-core processors. As the number of cores increases, more contention can occur on the communication bus which raises a bottleneck of the overall performance. Therefore, in order to reduce contention incurred on the communication bus, a many-core processor typically employs a Network-on-Chip (NoC) to achieve data-exchange. Real-time embedded systems have been widely utilized for decades. In addition to the correctness of functionalities, timeliness is also an important factor in such systems. Violation of specific timing requirements can result in performance degradation or even fatal problems. While executing real-time applications on many-core processors, the timeliness of a NoC, as a communication subsystem, is essential as well. Unfortunately, many real-time system designs over-provision resources to guarantee the fulfillment of timing requirements, which can lead to significant resource waste. For example, analysis of a NoC design yields that the network is already saturated (i.e. accepting more traffic can incur requirement violation), however, in reality the network actually has the capacity to admit more traffic. In this thesis, we target such resource wasting problems related to design and analysis of NoCs that are used in real-time systems. We propose a number of solutions to improve the schedulability of real-time traffic over wormhole-switched NoCs in order to further improve the resource utilization of the whole system. The solutions focus mainly on two aspects: (1) providing more accurate and efficient time analyses; (2) proposing more cost-effective scheduling methods.

Populärvetenskaplig sammanfattning

I ett modernt industrisystem har kravet på beräkningskapacitet ökat dramatiskt. Detta för att möjliggöra för ett högre antal funktioner, för att behandla en större mängd data eller för att göra snabbare och säkrare beslut under körtid. För att klara de ökade beräkningskraven blir flerkärniga processorer idag en allt mer attraktiv lösning. På en processor med många kärnor kan ett stort antal program köras parallellt och på så sätt öka beräkningsprestandan. Network-on-Chip (NoC) är ett sammankopplingsmedium mellan kärnor på en parallell plattform, som vanligtvis används i flerkärniga processorer eller så kallade system-on-chips i allmänhet.

Inbyggda realtidssystem har använts i stor utsträckning i årtionden. Utöver funktionell korrekthet är timing också en avgörande faktor i sådana system. Avvikelse från uppställda tidskrav kan leda till försämrad prestanda eller till och med skada med dödlig utgång. Samtidigt som man kör realtidsapplikationer på processorer med många kärnor, är det också viktigt att ett kommunikationssystem som NoC uppfyller krav på timing. Många designers av realtidssystem överdimensionerar mängden resurser som krävs för att garantera uppfyllandet av tidskrav. Denna designstrategi är säker med avseende på att uppfylla tidskrav men kan leda till betydande underutnyttjande av resurser. Denna avhandling riktar in sig på detta potentiella underutnyttjande av resurser i samband med design och analys av det NoC som används i realtidssystem. I avhandlingen föreslår vi flera lösningar för att förbättra schemalaggningsen av realtidstrafik över så kallat wormhole-switched NoC för att på så sätt förbättra resursutnyttjandet av hela systemet. De föreslagna lösningarna fokuserar huvudsakligen på utvecklandet av två aspekter: (1) mer exakta och effektiva tid-sanalyser; samt (2) mer kostnadseffektiva schemaläggningsmetoder.

Resultaten som presenteras i avhandlingen visar på signifikant förbättring av både analys och schemaläggning av realtidskommunikation över två vanliga typer wormhole-switched NoCs.

Abstract

In a modern industrial system, the requirement on computational capacity has increased dramatically, in order to support a higher number of functionalities, to process a larger amount of data or to make faster and safer run-time decisions. To cope with the increased computational requirements, many-core processors are gaining more and more attention nowadays. On a many-core processor, a large number of software programs can be executed in parallel, which can thus boost the computational performance. The Network-on-Chip (NoC) is an interconnection medium between intellectual property cores on a massively parallel platform, which is commonly used in many-core processors or system-on-chips in general.

Real-time embedded systems have been widely utilized for decades. In addition to the correctness of functionalities, timeliness is also an important factor in such systems. Violation of specific timing requirements can result in performance degradation or even fatal problems. While executing real-time applications on many-core processors, the timeliness of a NoC, as a communication subsystem, is essential as well. Unfortunately, many real-time system designs over-provision resources to guarantee the fulfillment of timing requirements, which can lead to significant resource waste. In this thesis, we target such resource wasting problems related to design and analysis of NoCs that are used in real-time systems. We propose a number of solutions to improve the schedulability of real-time traffic over wormhole-switched NoCs in order to further improve the resource utilization of the whole system. The solutions focus mainly on two aspects: (1) providing more accurate and efficient time analyses; (2) proposing more cost-effective scheduling methods.

The results presented in the thesis show a significant improvement both with regards to analysis and scheduling of real-time traffic in the context of two commonly used wormhole-switched NoC designs.

To my family

Acknowledgments

First of all, I would like to thank my supervisors Prof. Thomas Nolte and Dr. Moris Behnam. Without your inspirations, guidance and encouragement, I would not be able to produce this thesis. I also would like to express my appreciation to Prof. Luis Almeida and Prof. Shinpei Kato for the inspirations and cooperations, and Prof. Björn Lisper for reviewing my thesis and providing helpful comments.

Many thanks go to my colleague Matthias Becker for the close discussions and helps. I would also like to thank all former and present colleagues from my research group, Prof. Kristian Sandström, Prof. Reinder J. Bril, Dr. Alessandro Papadopoulos, Dr. Hang Yin, Dr. Mikael Åsberg, Dr. Mohammad Ashjaei, Dr. Nima Khalilzad, Dr. Rafia Inam, Dr. Saad Mubeen, Sara Afshar, Hamid Reza Faragardi, Daniel Hallmans, for inspiring discussions and cooperations.

Further more, I would like to thank all the professors and lecturers at MDH, Dr. Antonio Cicchetti, Prof. Emma Nehrenheim, Prof. Erik Dahlquist, Prof. Gordana Dodig-Crnkovic, Prof. Hans Hansson, Dr. Hongyu Pei-Breivold, Prof. Jan Gustafsson, Dr. Margaret Obondo and Dr. Monica Odlare, from whom I have learned a lot during my graduate study.

Also, I would like to thank all the administrative stuff at IDT, especially Carola, Sofia and Susanne for your helps.

Besides the research work and study, I also had a lot fun during conference trips, badminton, barbecue, carting, cinema, etc. Many thanks also go to all the fun-mates: Abhilash, Adnan, Aida, Alessio, Andreas, Aneta, Ashalatha, Batu, Cristina, Dag, Eduard, Elena, Filip, Francisco, Gabriel, Gregory, Guillermo, Irfan, Jakob, Juraj, Kan, Kivanc, LanAnh, Leo, Luka, Mehrdad, Mirgita, Nandinbaatar, Nesredin, Nils, Omar, Pablo, Patrick, Per, Predrag, Raluca, Sara Ab., Sara x. Ab., Severine, Simin, Svetlana, and others!

Finally, I would like to express my very great appreciation to my family for always encouraging and supporting.

This work has been supported by the Swedish Research Council (Vetenskapsrådet) under the project START and KK-stiftelsen through the project PREMISE.

Meng Liu
Västerås, January, 2017

List of publications

Papers included in the thesis¹

Paper A: *Tighter Time Analysis for Real-Time Traffic in On-Chip Networks with Shared Priorities*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 10th IEEE/ACM International Symposium on Networks-on-Chip (NOCS), 2016.

Paper B: *Improved Priority Assignment for Real-Time Communications in On-Chip Networks*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 23rd International Conference on Real-Time Networks and Systems (RTNS), 2015.

Paper C: *A Dependency-Graph Based Priority Assignment Algorithm for Real-Time Traffic over NoCs with Shared Virtual-Channels*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 12th IEEE World Conference on Factory Communication Systems (WFCS), 2016.

Paper D: *Scheduling Real-Time Packets with Non-Preemptive Regions on Priority-based NoCs*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 22th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), 2016.
(Best Student Paper Award)

Paper E: *A Tighter Recursive Calculus to Compute the Worst-Case Traversal Time of Real-Time Traffic over NoCs*, Meng Liu, Matthias Becker, Moris

¹The included articles have been reformatted to comply with the thesis layout.

Behnam, Thomas Nolte. In Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), 2017.

Paper F: *Buffer-Aware Analysis for Worst-Case Traversal Time of Real-Time Traffic over RRA-based NoCs*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), 2017.

Paper G: *Using Segmentation to Improve Schedulability of RRA-based NoCs with Mixed Traffic*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), 2017.

Additional papers, not included in the thesis

1. *Response Time Analysis for Static Priority Based SpaceWire Networks*, Meng Liu, Moris Behnam, Thomas Nolte, Luis Almeida. In Proceedings of the 2nd International Workshop on Worst-Case Traversal Time (WCTT), 2012.
2. *Worst-Case Delay Analysis of Master-Slave Switched Ethernet Networks*, Mohammad Ashjaei, Meng Liu, Moris Behnam, Ahlem Mifdaoui, Luis Almeida, Thomas Nolte. In Proceedings of the 2nd International Workshop on Worst-Case Traversal Time (WCTT), 2012.
3. *An EVT-based Worst-Case Response Time Analysis of Complex Real-Time Systems*, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 8th IEEE International Symposium on Industrial Embedded Systems (SIES), 2013.
4. *Applying the Peak Over Thresholds Method on Worst-Case Response Time Analysis of Complex Real-Time Systems*, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 19th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), 2013.
5. *Schedulability Analysis of Mixed-queued Controller Area Networks with Multi-Frame Messages*, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 18th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), WIP, 2013.

6. *Probabilistic Application Interfaces for Hierarchical Scheduling*, Nima Khalilzad, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 34th IEEE Real-Time Systems Symposium (RTSS), WIP, 2013.
7. *Schedulability Analysis of GMF-Modeled Messages over Controller Area Networks with Mixed-Queues*, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 10th IEEE International Workshop on Factory Communication Systems (WFCS), 2014.
8. *Challenges with Probabilities in Response-Time Analysis of Real-Time Systems*, Thomas Nolte, Meng Liu, Björn Lisper. In Proceedings of the 5th International Real-Time Scheduling Open Problems Seminar (RT-SOPS), 2014.
9. *An Adaptive Server-Based Scheduling Framework with Capacity Reclaiming and Borrowing*, Meng Liu, Moris Behnam, Shinpei Kato, Thomas Nolte. In Proceedings of the 20th IEEE International Conference on embedded and Real-Time Computing Systems and Applications (RTCSA), 2014.
10. *A Server-based Approach for Overrun Management in Multi-Core Real-Time Systems*, Meng Liu, Moris Behnam, Shinpei Kato, Thomas Nolte. In Proceedings of the 19th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2014.
(IEEE Industrial Electronics Society Scholarship Award)
11. *A Stochastic Response Time Analysis for Communications in On-Chip Networks*, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 21st IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), 2015.
12. *Adaptive Routing of Real-Time Traffic on a 2D-Mesh Based NoC*, Matthias Becker, Meng Liu, Moris Behnam, Thomas Nolte. In Proceedings of the 21st IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), WIP, 2015.
13. *Towards Stochastic Response Time Analysis for CAN Messages with Multiple Probabilistic Factors*, Meng Liu, Saad Mubeen, Moris Behnam, Thomas Nolte. In Proceedings of the 21st IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), WIP, 2015.

14. *On Providing Real-Time Guarantees in Cloud-based Platforms*, Meng Liu, Cezar Chiru, Moris Behnam, Kristian Sandström, Thomas Nolte. In Proceedings of the 12th IEEE World Conference on Factory Communication Systems (WFCS), WIP, 2016.
15. *Using Segmentation to Improve Schedulability of Real-Time Packets on NoCs with Mixed Traffic*, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 14th International Workshop on Real-Time Networks (RTN), 2016.

Contents

I	Thesis	1
1	Introduction	3
1.1	Research Methodology	5
1.2	Goal of the Thesis	6
1.3	Thesis Overview	8
2	Background	9
2.1	Real-Time Systems	9
2.2	Wormhole-Switched Network-on-Chips	10
2.3	Topologies of NoCs	11
2.4	Access Control in NoCs	11
2.5	Time Analysis	15
3	Technical Contributions	17
4	Conclusions	23
4.1	Summary and Conclusions	23
4.2	Future Work	24
5	Overview of the Papers	27
5.1	Paper A	27
5.2	Paper B	27
5.3	Paper C	28
5.4	Paper D	28
5.5	Paper E	29
5.6	Paper F	29
5.7	Paper G	30

Bibliography 31

II Included Papers 37

**6 Paper A:
Tighter Time Analysis for Real-Time Traffic in On-Chip Networks
with Shared Priorities** 39

6.1 Introduction 41
 6.1.1 Contribution 42
6.2 Related Work 42
6.3 System Model 43
6.4 Recapitulate the Existing RTA 44
6.5 Improved Time Analysis 47
 6.5.1 Pessimism in Computed Blocking 47
 6.5.2 Pessimism in Computed Interference 52
6.6 Evaluation 55
 6.6.1 Case Study 55
 6.6.2 General Evaluation 56
6.7 Conclusion and Future Work 59
Bibliography 61

**7 Paper B:
Improved Priority Assignment for Real-Time Communications in
On-Chip Networks** 65

7.1 Introduction 67
7.2 Related Work 69
7.3 System Model 70
7.4 Recapitulate the Heuristics-based Priority Assignment Algorithm 71
 7.4.1 Lower and Upper Bound Analysis 71
 7.4.2 The Heuristic Search Algorithm 72
 7.4.3 Missing Cases of The HSA 73
 7.4.4 Inefficient Backtracking under The HSA 75
7.5 Improved Priority Assignment of NoC Flows 77
7.6 Evaluation 85
 7.6.1 Experiment Result: Schedulability Ratio 86
 7.6.2 Experiment Result: Number of Assignments 88
7.7 Conclusion and Future Work 91
Bibliography 93

8	Paper C:	
	A Dependency-Graph Based Priority Assignment Algorithm for Real-Time Traffic over NoCs with Shared Virtual-Channels	97
8.1	Introduction	99
8.1.1	Contribution	100
8.1.2	Related Work	100
8.2	System Model	102
8.3	Priority Assignment of NoC Flows with Priority Sharing	103
8.3.1	Improved Schedulability Test	103
8.3.2	The eGHTSA	107
8.4	Evaluation	112
8.4.1	Case Study	112
8.4.2	General Evaluation	114
8.5	Conclusion and Future Work	117
	Bibliography	119
 9	 Paper D:	
	Scheduling Real-Time Packets with Non-Preemptive Regions on Priority-based NoCs	123
9.1	Introduction	125
9.1.1	Contributions	126
9.1.2	Related Work	127
9.1.3	Organization	128
9.2	Network Model	128
9.3	Transmission Policy	129
9.4	Recapitulate the RTA for Flit-Level Preemptive NoCs	130
9.5	Scheduling NoC Packets with Non-preemptive Regions	132
9.5.1	Extended RTA of Packets with Non-Preemptive Regions	132
9.5.2	Computing the Blocking Tolerance	135
9.5.3	Selecting the Lengths of Non-Preemptive Regions	138
9.6	Evaluation	142
9.6.1	Analysis Based Evaluation	142
9.6.2	Simulation Based Evaluation	145
9.6.3	Case Study	146
9.7	Conclusion and Future Work	147
	Bibliography	149

10 Paper E:

A Tighter Recursive Calculus to Compute the Worst-Case Traversal Time of Real-Time Traffic over NoCs **153**

10.1 Introduction 155

 10.1.1 Contribution 156

 10.1.2 Related Work 156

10.2 System Model 158

10.3 Recapitulation of RC and BP/BPC 159

10.4 Tighter Recursive Calculus 162

10.5 Evaluation 169

 10.5.1 Comparison between TRC and BP/BPC 169

 10.5.2 Comparison between TRC and RC 172

10.6 Conclusion and Future Work 173

Bibliography 175

11 Paper F:

Buffer-Aware Analysis for Worst-Case Traversal Time of Real-Time Traffic over RRA-based NoCs **179**

11.1 Introduction 181

 11.1.1 Contribution 182

 11.1.2 Related Work 182

11.2 System Model 183

11.3 Recapitulation of RC 186

11.4 Optimistic Results in RC 188

11.5 Revised Recursive Calculus 189

 11.5.1 Removing Optimism 189

 11.5.2 Supporting Packetization 195

11.6 Evaluation 198

11.7 Conclusion and Future Work 200

Bibliography 203

12 Paper G:

Using Segmentation to Improve Schedulability of RRA-based NoCs with Mixed Traffic **207**

12.1 Introduction 209

 12.1.1 Related Work 210

12.2 System Model 211

12.3 Segmentation for NoCs with Real-Time Traffic 212

12.4 Segmentation for NoCs with Mixed Traffic 217

12.5	Evaluation	219
12.5.1	Evaluation of Segmentation on NoCs with Real-Time Traffic	219
12.5.2	Evaluation of Segmentation on NoCs with Mixed Traffic	223
12.6	Conclusion and Future Work	225
	Bibliography	227

I

Thesis

Chapter 1

Introduction

In a modern industrial system, the requirement on computational capacity has increased dramatically, in order to support a higher number of functionalities, to process a larger amount of data or to make faster and safer run-time decisions. In a traditional single-core processor, threads can only be executed sequentially (i.e. only one thread can be executed at a time). The only way to improve the processing throughput on such type of processors is to run the cores faster, which on the other hand increases power consumption and hardware cost. Therefore, in the mid-2000s, processor vendors such as Intel and AMD emphasized multi-core processors as the answer to scaling system performance. In a multi-core processor, a small number of cores (typically up to eight) are integrated in the same silicon typically connected by an on-chip communication bus. Software programs can be executed in parallel on such type of processors, which can thus boost the computational performance. Many-core processors are specialized multi-core processors with a larger number of cores which are designed to achieve a higher degree of parallel processing. An on-chip communication bus is a central intersection used for data-exchange between cores, memory and I/O in most multi-core processors. As the number of cores increases, more contention can be incurred on the communication bus which raises a bottleneck of the overall performance. Therefore, in order to reduce contention incurred on the communication bus, a many-core processor typically employs a Network-on-Chip (NoC) to achieve data-exchange. In this thesis, we focus on wormhole-switched [1] NoCs which caters the needs of integrated circuit design [2].

Real-time embedded systems have been widely utilized for decades. In

addition to the correctness of functionalities, timeliness is also an important factor in such systems. Violation of specific timing requirements can result in performance degradation or even fatal accidents. Therefore, in order to guarantee the correct behavior of such systems, system designers need to carefully analyze that all the timing requirements can be fulfilled. One of the most typical timing requirements is regarding schedulability which refers to determining if a real-time task¹ can be executed within a specific time interval. Such type of requirement exists in many time-critical systems, such as flight-control systems on aircrafts, brake-by-wire systems in vehicles, or automatic control systems in factories. In order to verify the satisfaction of the requirement on schedulability, analyzing the timing behavior of each task is of importance. In a real-time system using multi/many-core processors, the processing time of a task typically depends on (1) the execution time that is used to execute the code in computational cores, and (2) the transmission time that is used for data-exchange between cores, memory and I/O over on-chip networks. To provide reliable estimate of processing times, the predictability of both factors is essential. Regarding the first factor, many research works have been conducted in the past decades. In this thesis, we focus on the second factor which is about real-time communication over NoCs.

A typical solution to guarantee the fulfillment of timing requirements is to analyze the timing behavior of a real-time system and then to reserve sufficient system resources such that execution of tasks can be completed timely. Unfortunately, many real-time systems designs over-provision resources which can lead to significant resources waste. For example, a particular NoC design guarantees the schedulability of the network, however it results in a low bandwidth utilization. Similarly, analysis of a NoC design yields that the network is already saturated (i.e. accepting more traffic can cause requirement violation), however, in reality the network actually has the capacity to admit more traffic. In this thesis, we target such resource wasting problems related to design and analysis of NoCs that are used in real-time systems. We propose several solutions to improve the schedulability of real-time traffic over wormhole-switched NoCs in order to further improve the resource utilization of the whole system. The solutions focus mainly on two aspects: (1) providing more accurate and efficient time analyses; (2) proposing more cost-effective scheduling methods.

¹A task is a basic unit of software programs.

1.1 Research Methodology

In this thesis, we develop different approaches to analyze the timing property of wormhole-switched NoCs that can be used by system designers to verify the fulfillment of real-time requirements. Additionally, we also propose mechanisms to allow more workload admitted in a NoC without raising hardware cost. Therefore, the main research contributions of this thesis can be identified as *techniques* [3] to enable service-guaranteed and cost-effective real-time communication over wormhole-switched NoCs. As discussed in [3], the approach or method used to conduct such type of research include:

Invent new ways to do some tasks, including procedures and implementation techniques. Develop a technique to choose among alternatives.

Three types of research results along with their criteria of quality have been discussed in [3] which are originally identified by Brooks in [4]:

- *Findings: well-established scientific truths - judged by truthfulness and rigor;*
- *Observations: reports on actual phenomena - judged by interestingness;*
- *Rules-of-thumb: generalizations, signed by an author (but perhaps not fully supported by data) - judged by usefulness.*

The solutions proposed in this thesis are validated according to the above criteria.

The research process to conduct this thesis is illustrated in Figure 1.1. The process starts from a general research goal which is further divided into a number of subgoals. To approach each subgoal, several research questions are raised accordingly. We then perform a literature review to study the state-of-the-art approaches. The literature review can bring us insights regarding benefits of the existing techniques as well as their limitations. A number of problems can then be formulated such as how to apply a good technique in another context to benefit other disciplines, or how to solve problems that are not covered by the existing solutions due to their limitations. Different solutions targeting concrete problems are then proposed and developed. Formal proofs are provided to prove the correctness of the solutions. The proposed techniques are evaluated using simulations, synthetic experiments and case studies. Finally, we present our results in scientific publications where the works are peer

reviewed by other researchers. Based on the conclusions from the committed work and the feedback from peer reviews, we can revisit the research goals and research questions. This cycle is repeated to continuously conduct our research works.

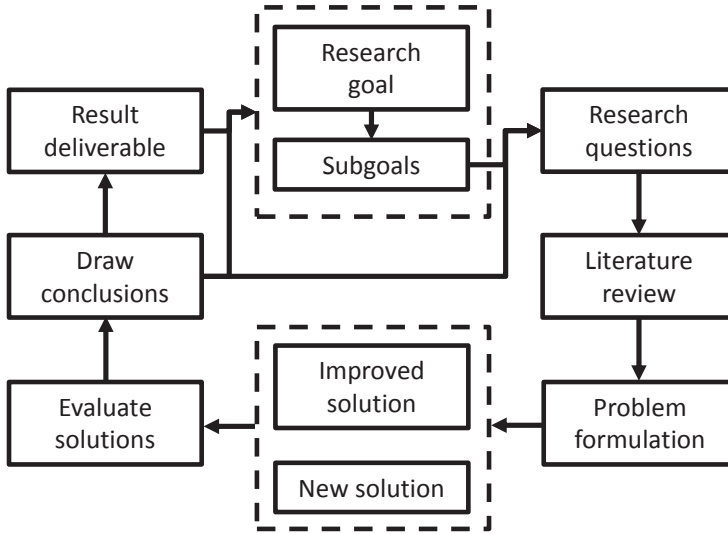


Figure 1.1: The flow of research process.

1.2 Goal of the Thesis

Nowadays many-core processors are gaining more and more attention due to their high computational capability along with effective hardware cost. In order to utilize many-core processors in real-time systems, the timeliness of executions in the processors is of importance. As a communication subsystem which interconnects cores, memory and I/O in a many-core processor, the timing behavior of a NoC needs to be predictable during system design. One of the most important timing requirements is regarding schedulability, which refers to determining if a real-time packet² can be delivered within a specific time duration (called *deadline*).

²A packet is a formatted unit of data in a network.

Most of the existing real-time system designs over-provision resources to guarantee the fulfillment of timing requirements, which can lead to significant resource waste. In this thesis, we target such resource wasting problems for NoCs used in real-time systems. Under such a target, we define the main research goal as follows.

To improve the schedulability of a NoC which further improves the resource utilization of the whole system.

Schedulability can typically be improved via two aspects:

- providing more accurate and efficient time analyses;
- proposing more cost-effective scheduling methods.

Time analysis is a commonly utilized technique to verify the schedulability of a network or system. In the context of NoCs, targeting hard real-time applications, a proper time analysis computes an upper-bound of the transmission time for each packet. If the computed estimate is no larger than the given deadline, this packet is guaranteed to be schedulable. On the other hand, if the estimate is larger than the deadline, this packet is considered as unschedulable which makes the system design become invalid. In order to provide safe analysis results along with low computational complexity, most of the existing analysis methods tend to use approximations. As a result, the analyses can involve significant pessimism (i.e. the computed estimate is much larger than the actual transmission time). If we can reduce the pessimism involved in the analyses, the analysis results can become more practical. Consequently, we can allow a specific network to admit more real-time traffic which can thus improve the network utilization.

Proposing methods to guide the design of a NoC is another typical way to improve schedulability. The improvement on schedulability can be achieved through *modifying communication mechanisms of the network* (e.g. scheduling mechanisms) or *changing certain characteristics of the traffic* (e.g. priorities or packet sizes).

Considering distinct features of different NoC architectures, it is difficult to develop a unified solution that can be generally applied on all NoC designs. This thesis focuses on NoCs with two types of scheduling mechanisms, being *fixed-priority based* and *round-robin based* NoCs. Aiming at the above research goal, different solutions targeting each type of NoC are proposed.

According to the above categorization, the main research goal can be formulated into the following four subgoals which are used as a guideline of our research:

Subgoal I (SG-I): Providing a safe, tight and efficient time analysis of fixed-priority based NoCs.

Subgoal II (SG-II): Proposing more cost-effective scheduling methods for fixed-priority based NoCs.

Subgoal III (SG-III): Providing a safe, tight and efficient time analysis of round-robin based NoCs.

Subgoal IV (SG-IV): Proposing more cost-effective scheduling methods for round-robin based NoCs.

1.3 Thesis Overview

In Chapter 2, we briefly introduce the background knowledge of the works included in the thesis. In Chapter 3, we present the main technical contributions of this thesis with respect to the corresponding research goals. A summary of the included works is presented in Chapter 4, together with some prospects of the future work. In Chapter 5, we present an overview of all the included papers, and these papers are presented in Chapter 6-12.

Chapter 2

Background

2.1 Real-Time Systems

A real-time system is a computer system, where the correct behavior depends on both functional correctness and timeliness. In such systems, predictability is an important property that can allow system designers to be able to analyze the timing behavior correctly. For an acceptable system design, all the timing requirements need to be fulfilled. Regarding different levels of requirement satisfaction, real-time systems can be categorized into two types: *hard* real-time systems and *soft* real-time systems.

In a hard real-time system, all the timing requirements must be strictly satisfied. A violation of the requirements can result in catastrophic consequences. For example, in an automotive system, the brake-by-wire subsystem and airbag subsystem are typical hard real-time systems. Violations of timing requirements in such systems can cause personal and property losses.

In a soft real-time system, certain violations of timing requirements are tolerable. In such systems, requirement violations may degrade the system performance, but cannot cause fatal problems. For example, a multimedia entertainment system is a soft real-time system, since the requirement violation may only degrade the user satisfaction without causing any serious impact.

2.2 Wormhole-Switched Network-on-Chips

The Network-on-Chip (NoC) is an interconnection medium between intellectual property cores on a massively parallel platform, which is commonly used in a many-core processor or a System-on-Chip (SoC) in general. A many-core platform is typically arranged into a number of nodes, where each node contains one or multiple cores as well as local memory. Each node has a router that connects to all local cores. The routers in different nodes are in turn connected to each other and they thus comprise the NoC. An example of a NoC is shown in Figure 2.1. Compared to conventional bus and crossbar based interconnections, NoCs can achieve notably better scalability and power efficiency [5].

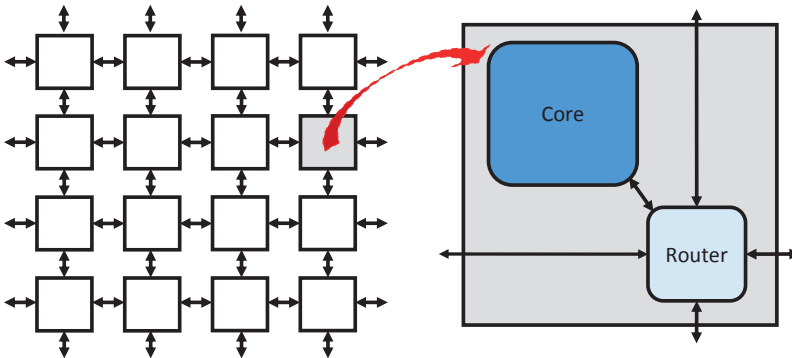


Figure 2.1: A node in a 2D-meshed NoC.

Wormhole-switching [1] is a commonly used technique for NoCs [2]. Under such a mechanism, each packet is divided into a number of flow control digits (called *flits*) which are the elementary units of transmission. Once a router receives one flit of a packet, it can directly transmit the flit without waiting for the arrival of the complete packet. A header flit, which contains the routing information and packet size, is transmitted first in the network. As long as the next link on the path is free and the buffer on the next router can accommodate at least one flit, the header continues its transmission. As the header flit advances along its specific route, the remaining flits follow in a pipelined manner. During this transmission, the flits of one packet can span over multiple routers,

hence the name wormhole-switching. Compared to a store-and-forward mechanism, wormhole-switching requires significantly smaller buffers, which suits the design principles of NoCs. Moreover, wormhole-switching can reduce the transmission latency of a packet compared to a store-and-forward mechanism. For a store-and-forward mechanism the latency (assuming that there is no competing traffic) is proportional to the product of the packet size and the number of hops, while for wormhole-switching the latency is proportional to the product of the flit size and the number of hops.

2.3 Topologies of NoCs

The topology of a NoC depicts how the nodes in the network are connected. A number of topologies have been proposed in the literature, such as 2D mesh which is a basic grid architecture (see Figure 2.2-a), 2D torus which is based on 2D mesh with extra links connecting nodes along edges (see Figure 2.2-b), 3D hypercube which consists of multiple layers of 2D meshes (see Figure 2.2-c) and octagon where all the nodes are fully interconnected (see Figure 2.2-d) [5].

This thesis focuses on the 2D-mesh based topology. Because of the simplicity of implementation and predictability of transmissions, such a topology has been widely utilized in many research works (e.g. [6][7][8]) as well as commercial implementations (e.g. [9][10]). In a 2D-mesh based NoC, a router has up to 5 ports which are used to connect other components. As shown in Figure 2.1, a router has one port connected to the local core and four ports (to the north, the east, the south and the west respectively) connected to other routers.

2.4 Access Control in NoCs

In order to provide service guarantees to real-time traffic, the predictability of a NoC is necessary. Since a NoC is shared among multiple real-time traffic flows¹, the access to the network needs to be under control. A number of access control mechanisms for NoCs have been proposed in the literature, such as Time-Division Multiplexing (TDM) based, Fixed Priority (FP) based and Round-Robin (RR) based approaches.

¹A flow is a series of packets with the same characteristics.

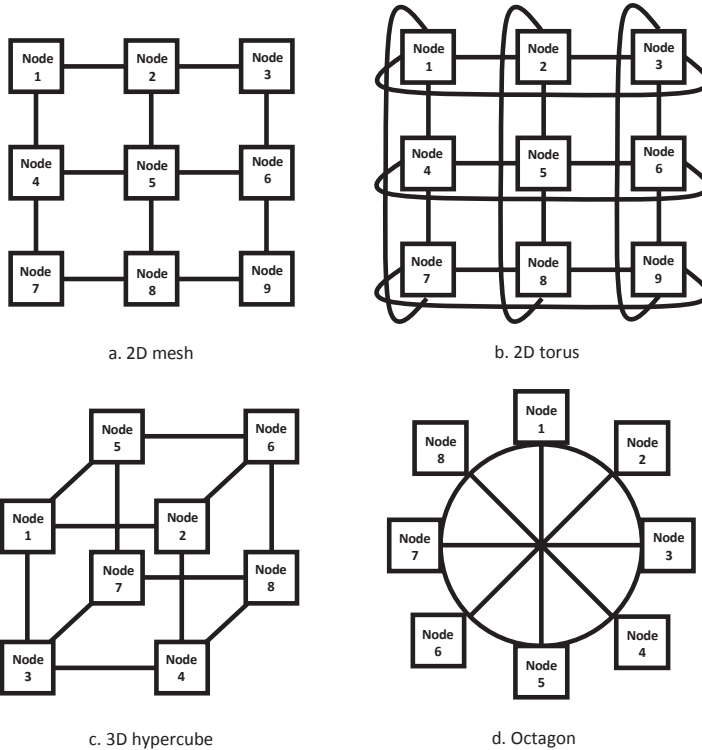


Figure 2.2: Examples of NoC topologies.

In a TDM-based NoC, the schedule of transmission is accomplished through pre-defined time-slots. Each packet can only use its assigned time-slot for its transmission. Under such a mechanism, the transmission of different packets can be isolated from each other. Consequently, this can also accommodate both best-effort traffic which does not have any timing requirements and real-time traffic which is time-critical in the same network. This type of NoC has been considered and implemented in many existing works, such as *Æthereal* [11] and time-triggered NoCs [12][13].

In a FP-based NoC, the network access is controlled by priorities of packets. Each packet is assigned a fixed priority. When two packets compete for the access to the same output-port, the packet with higher priority will be transmitted first. If a NoC supports a preemptive scheme, the transmission of a packet with a higher priority can preempt the transmission of another packet with a

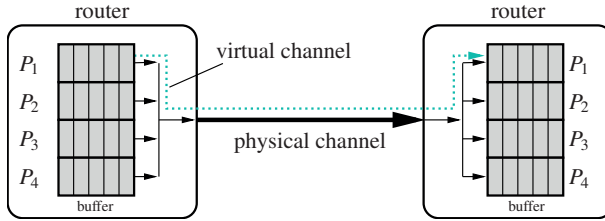


Figure 2.3: Virtual channels on the NoC router.

lower priority. FP-based NoCs typically use Virtual-Channels (VCs) [14] to support preemptions between different priority levels. A VC is an additional buffer connected to a physical channel. The transmission of each packet can only use its assigned VC, thus packets using different VCs can be stored in separate buffers. While one packet is blocked, a second packet with the same path but a different VC can still proceed, since the physical channel is idle and the buffers are not shared. The dotted line in Figure 2.3 depicts such a virtual-channel. Each VC is associated with a specific priority level. A packet using a VC with a higher priority can preempt the transmission of a packet transmitted in a VC with a lower priority. This type of NoC has been studied in many research works such as [6][15][16][17][8][18][19].

In a RR-based NoC, all the packets are treated in a fair manner. One example of a RR-based NoC is shown in Figure 2.4. Under such a mechanism, each input buffer can deliver at most one packet to the output link within one round-robin cycle. The transmission of each packet is commonly non-preemptive. This type of NoC design has been utilized in both academic research (e.g. [20][21][22][23][24]) and commercial implementations (e.g. Tiler TILE64 [9], Kalray MPPA[®] 256 [25], Adapteva Epiphany E64G401 [10]).

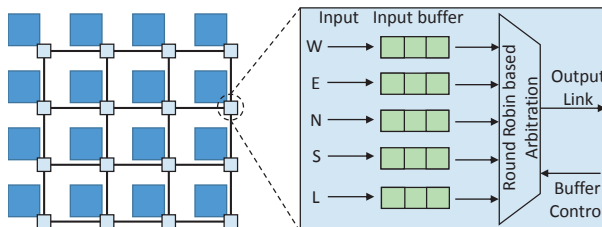


Figure 2.4: The abstracted architecture of a RR-based NoC router.

A general comparison of the above access control schemes is illustrated in Figure 2.5. A TDM-based approach has strong time predictability and hardware efficiency (i.e. hardware cost). However, since the schedule is predefined off-line, the runtime flexibility of a TDM-based approach is weaker than the others. A FP-based scheme achieves medium time predictability and runtime flexibility. However, due to the requirement of virtual-channels, the hardware efficiency of a FP-based scheme can be low. A RR-based design brings the highest runtime flexibility since it is independent on the characteristics of transmitted traffic. On the other hand, since a RR-based NoC treats all packets in the same manner, it is difficult to provide timing guarantees on specific traffic flows. Furthermore, developing a tight and efficient time analysis for RR-based NoCs is also challenging due to the nature of scheduling decisions. In summary, each of these schemes has its own advantages and disadvantages. The selection of access control mechanisms depends more on the requirements of applications. Recently, the authors in [26] present a detailed comparison between TDM-based and RR-based NoCs regarding bandwidth, latency, hardware cost and support of mixed-traffic, where their respective advantages and disadvantages are illustrated. In this thesis, we focus on FP-based and RR-based NoCs.

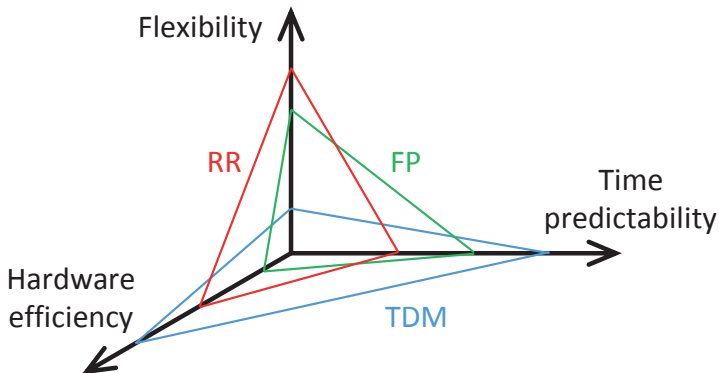


Figure 2.5: A comparison of access control schemes.

2.5 Time Analysis

Time analysis is a mathematical technique used to achieve a schedulability test, which is utilized during the design phase of a real-time system. In the context of NoCs, the results of a time analysis typically include estimates of transmission times. By comparing the computed estimates with the given deadlines, system designers can verify the schedulability of the whole network. Regarding fixed-priority based NoCs, most of the existing analyses (e.g. [6][15][27][8][18][19]) are based on the well-known Response Time Analysis [28] which is originally developed for task scheduling. For round-robin based NoCs, several different types of analysis methods have been proposed, such as Recursive Calculus [20] based approaches (e.g. [21][22][29]), Network Calculus [30] based approaches (e.g. [31][32]), Compositional Performance Analysis (CPA) [33] based approaches (e.g. [23]), queueing theory based approaches (e.g. [34][35]), and machine learning based approaches (e.g. [36]). The contributions presented in this thesis are more related to the Response Time Analysis based and the Recursive Calculus based analysis approaches.

Chapter 3

Technical Contributions

This thesis presents five main contributions targeting the formulated research goals.

Research Contribution 1: Developing a tighter time analysis for fixed-priority based NoCs with shared virtual-channels.

- **Introduction:** The NoC is the preferred interconnection medium for massively parallel platforms. Targeting real-time applications, fixed-priority based NoCs with virtual-channels have been proposed as a promising solution. In order to verify if specific time requirements can be satisfied, schedulability tests are typically used. Several analysis approaches have been proposed targeting priority-based NoCs. However, due to the approximation considered in the analyses, the results may involve a large amount of pessimism. The applicability of the analyses is thus limited in practice. In this work, we identify a number of properties of NoCs with shared priorities. An improved time analysis is proposed where pessimism in the analysis can be significantly reduced for many cases. In order to evaluate the proposed analysis, a number of experiments have been generated along with a case study based on an automotive application. The evaluation results show that the proposed analysis provides tighter estimates for around 70% of all the generated traffic flows.
- Targeting research subgoal: SG-I
- Included papers: Paper A

Research Contribution 2: Proposing priority assignment algorithms for traffic flows in fixed-priority based NoCs to improve schedulability of the network.

- Introduction: Fixed-priority based preemptive scheduling using virtual-channels is a solution to support real-time communications in on-chip networks. However, the different characteristics of NoCs compared to the single-core processor scheduling problem prevents the usage of the known optimal algorithms (e.g. the Audsley's algorithm) to assign priorities to messages. A heuristic search algorithm based approach (called HSA) focusing on the priority assignment for on-chip communications has been presented in the literature. HSA is much faster than an exhaustive search based solution, with a price of missing certain schedulable cases (i.e. non-optimal). In this work, we present two undirected-graph based priority assignment algorithms, GESA and GHSA. In contrast to the previous work, we can decrease the search space significantly by taking the interference dependencies of different messages on the network into account. A number of experiments are generated, in order to evaluate the proposed algorithms. The results show that GESA can always achieve higher schedulability ratios than HSA, but may require longer processing time. On the other hand, GHSA has the same performance as HSA regarding the schedulability, but can significantly improve the efficiency. For example, when the network contains 50 flows, GHSA is 3.73 times faster than HSA in average, and GHSA is even 66 times faster than HSA for certain cases. An extension of GHSA (called eGHSA) has also been proposed targeting NoCs with shared virtual-channels. Such type of NoC architecture is more practical since the incurred buffer cost is lower than the platforms considered in HSA and GHSA.
- Targeting research subgoal: SG-II
- Included papers: Paper B and C

Research Contribution 3: Introducing non-preemptive regions to NoC packets to improve the overall schedulability.

- Introduction: Most of the existing NoC implementations which can support fixed-priority based scheduling use a flit-level preemptive scheduling. Under such a mechanism, preemptions can happen between the transmissions of successive flits. In this work, we present a modified

framework where the non-preemptive region of each NoC packet increases from a single flit. Using the proposed approach, the response times of certain packet flows can be reduced, which can thus improve the schedulability of the whole network. As a result, the utilization of NoCs can be improved by admitting more real-time traffic. Schedulability tests regarding the proposed framework are presented along with the proof of the correctness. Moreover, a number of experiments as well as a case study based on an automotive application have been generated. Compared to the original flit-level preemptive scheduling, the proposed solution EDBT provides slight improvement (up to 3%) regarding schedulability ratio, and the proposed HPDBT achieves higher improvement (up to 10%).

- Targeting research subgoal: SG-II
- Included papers: Paper D

Research Contribution 4: Developing a tight and efficient time analysis for round-robin based NoCs.

- Introduction: In this work, we focus on a Round-Robin Arbitration (RRA) based wormhole-switched NoC which is a common architecture used in most of the existing implementations. In order to execute real-time applications on such a NoC based platform, the requirement on schedulability, which refers to if real-time packets can be delivered within the given time durations, has to be fulfilled. Timing analysis is a common tool to verify the schedulability of a real-time system. Unfortunately, the existing timing analyses of RRA-based NoCs either provide too pessimistic estimates which results in overly allocated resources, or have a high computational complexity which limits the applicability in reality. Therefore, we present an improved timing analysis, aiming to provide tighter estimates along with acceptable computation time. From the evaluation results, we can clearly observe the improvement achieved by the proposed time analysis.

Moreover, we observe that the original Recursive Calculus (RC), which has been utilized to analyze RRA-based NoCs in a number of related works, can produce optimistic (unsafe) estimates when the buffer at each router can hold more than one flit. However, router buffers in most of the existing NoC designs can actually hold multiple flits. In this case, the analysis results computed by RC will not be acceptable if the applications are time-critical. Therefore, we also propose a Revised Recursive

Calculus (RRC) which extends RC by considering buffer-effects as well as supporting packetization.

- Targeting research subgoal: SG-III
- Included papers: Paper E and F

Research Contribution 5: Proposing a segmentation algorithm for round-robin based NoCs to improve the schedulability of real-time traffic.

- Introduction: Most of the existing NoC designs focus on the performance with respect to average throughput, which makes them less applicable for real-time applications especially when applications have hard timing requirements on the worst-case scenarios. In this work, we propose a novel segmentation algorithm targeting RRA-based NoCs in order to improve the schedulability of real-time traffic without modifying the hardware architecture. Additionally, we also address the problem of transmitting both real-time traffic and best-effort traffic in the same NoC. The proposed solutions aim to provide timing guarantees to real-time traffic and achieve low latency for best-effort traffic. According to the evaluation results, the proposed segmentation solution can significantly improve the schedulability of the whole network.
- Targeting research subgoal: SG-IV
- Included papers: Paper G

The relation between the research subgoals and research contributions is depicted in Table 3.1.

	Main Goal			
	SG-I	SG-II	SG-III	SG-IV
Research Contribution 1 (Paper A)	√			
Research Contribution 2 (Paper B, C)		√		
Research Contribution 3 (Paper D)		√		
Research Contribution 4 (Paper E, F)			√	
Research Contribution 5 (Paper G)				√

Table 3.1: The relation between SGs and RCs.

My Contribution

I am the main contributor and the first author of all the included papers in this thesis. The presented works are performed in a close collaboration with my colleague Ph.D. candidate Matthias Becker who is involved in many detailed discussions on the proposed solutions, and my supervisors Prof. Thomas Nolte and Dr. Moris Behnam who contribute in reviewing the solutions and discussions.

Chapter 4

Conclusions

4.1 Summary and Conclusions

In this thesis, we have focused on improving the schedulability of real-time traffic over wormhole-switched on-chip networks. The target is accomplished from two perspectives: (1) providing more accurate time analyses; (2) proposing more cost-effective scheduling methods. Two types of NoC designs are considered in this thesis, being fixed-priority based and round-robin based NoCs.

Regarding the first perspective, we have developed a number of time analysis approaches targeting real-time traffic on the above two types of NoCs. In Paper A, we have proposed an improved time analysis for priority-based NoCs using shared virtual-channels. The evaluation results show that the proposed analysis involves less pessimism compared to the related works. In Paper E, we have presented a time analysis for round-robin based NoCs named Tighter Recursive Calculus (TRC). This solution improves an existing analysis method called Recursive Calculus (RC) by taking into account the arrival patterns of real-time traffic. According to the evaluation results, the proposed analysis can provide significantly tighter estimates compared to the original RC. On the other hand, compared to the algorithms called Branch and Prune (BP)/ Branch, Prune and Collapse (BPC), which can also produce tighter estimates, TRC has much lower computational complexity. Furthermore, we have identified an optimistic problem for RC due to lacking consideration of buffer-effects. As a result, the applicability of RC and other related analysis approaches can be strongly limited. Therefore, in Paper F, we have proposed a Revised Recursive

Calculus (RRC) which extends RC by addressing the above optimistic problem as well as supporting packetization of large packets.

In order to improve the schedulability of real-time traffic over NoCs, we have also presented a number of scheduling mechanisms to guide the network design. The proposed solutions either focus on modifying communication mechanisms or on changing traffic characteristics. In Paper B, we have presented two graph-based priority assignment algorithms for fixed-priority based NoCs. This work is later extended in Paper C to support priority-based NoCs with shared virtual-channels. The achieved improvement on both schedulability and efficiency can be observed from the evaluation results. In Paper D, we have applied the concept of limited preemptive scheduling on priority-based NoCs. We introduce non-preemptive regions to NoC packets, and we present how to select proper length of such regions. According to the evaluation results, using non-preemptive regions can always achieve higher schedulability ratio compared to the original flit-level preemptive transmission. In Paper G, we have applied a similar concept on round-robin based NoCs. In order to avoid modification on hardware, we retain the non-preemptive transmission mechanism as utilized in most round-robin based NoCs. Instead, we propose to use segmentation on NoC packets (i.e. dividing one NoC packet into a number of smaller sub-packets), and each sub-packet is transmitted in the same manner as normal NoC packets. We have presented algorithms to select proper sizes of sub-packets which can improve the overall schedulability of the network. A significant improvement achieved by the proposed solution is clearly shown in the evaluation results.

4.2 Future Work

There are a number of directions that can be considered for future works.

- In order to decrease computational complexity, most of the existing time analyses involve approximations especially when the NoC architecture is complicated. Consequently, these analyses always produce pessimistic results. We have developed a number of analysis methods with improved accuracy, however, there is still a lot of work remaining in this area. Recently, the authors in [19] have identified an optimistic problem of the existing time analysis for priority-based NoCs [6] because of lacking consideration of buffer-effects, and we have pointed out a similar problem for round-robin based NoCs in [37]. Even though the analysis presented in [37] has addressed such an optimistic problem, the solution on

the other hand incurs more additional pessimism in the analysis results. Form the industrial applicability point of view, a better time analysis with higher accuracy as well as acceptable computational complexity is necessary.

- All the analysis approaches proposed in this thesis focus on the estimation of the worst-case transmission time. These analyses are more suitable for hard real-time applications, where the performance depends on the worst-case scenarios. However, many applications do not have hard real-time constraints which are typically known as soft real-time applications. For example, the performance of certain systems may only depend on the average scenarios. For such type of applications, applying the proposed analyses can result in unnecessary over-dimensioned resource reservation. Therefore, targeting soft real-time applications, developing stochastic or statistical time analyses (e.g. [38][36][39]) is more appropriate.
- The priority assignment algorithms proposed in Paper B and C are based on the assumption that the sending and receiving tasks of each traffic flow are already allocated before starting the priority assignment process. However, the allocations of flow routes can actually affect the schedulability of the whole network. Therefore, it is better to combine the priority assignment problem together with the task mapping problem, which can potentially further improve schedulability. Similarly, the segmentation solution proposed in Paper G can also be combined with the task mapping problem to achieve even higher schedulability ratios.

Chapter 5

Overview of the Papers

5.1 Paper A

Tighter Time Analysis for Real-Time Traffic in On-Chip Networks with Shared Priorities, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 10th IEEE/ACM International Symposium on Networks-on-Chip (NOCS), 2016.

In this paper, we identify a number of properties to reduce pessimism involved in the existing time analysis of priority-based NoCs with shared virtual-channels. An improved analysis is then proposed based on the identified properties along with the proof of correctness. In order to evaluate the proposed analysis, a number of experiments as well as a case study based on an automotive application are generated. According to the evaluation results, the proposed Response Time Analysis (RTA) provides tighter estimates for around 70% of the tested flows compared to the original RTA. Significantly reduced pessimism can be observed from many cases as well. On the other hand, the proposed RTA requires more processing time as a price. However, according to the measurements, the extra time cost is quite acceptable in practice. This paper addresses Research Contribution 1.

5.2 Paper B

Improved Priority Assignment for Real-Time Communications in On-Chip Networks, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte.

In Proceedings of the 23rd International Conference on Real-Time Networks and Systems (RTNS), 2015.

In this paper, we point out the drawbacks of the existing heuristic priority assignment algorithm for on-chip communications (called HSA): missing cases and inefficiency. We then present two algorithms (named GHSA and GESA) for priority assignment of messages in wormhole-switched NoCs. In the proposed algorithms, we introduce an undirected-graph based search solution, where we take into account the dependencies between messages. Such a solution can safely decrease the search space thus improve the efficiency of the algorithms. A number of experiments have been generated. The results show that our proposed algorithm GHSA can be much faster than HSA while achieving the same schedulability ratio. On the other hand, GESA, which can achieve higher schedulability compared to HSA, is slower than HSA but much faster than an exhaustive search based solution. This paper addresses Research Contribution 2.

5.3 Paper C

A Dependency-Graph Based Priority Assignment Algorithm for Real-Time Traffic over NoCs with Shared Virtual-Channels, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 12th IEEE World Conference on Factory Communication Systems (WFCS), 2016.

In this paper, we present a dependency-graph based priority assignment algorithm targeting NoCs with shared priorities/virtual-channels. In the proposed algorithm, dependencies between flows are taken into account in order to avoid unnecessary schedulability tests. The efficiency of the priority assignment process can thus be improved. A number of experiments have been generated along with a case study, in order to evaluate the proposed algorithm compared to the existing approach. The results clearly show that while achieving the same number of required virtual-channels, the proposed algorithm (called eGHSA) is more efficient, especially when the network contains a large number of flows. This paper is a follow-up work of Paper B, thus it also addresses Research Contribution 2.

5.4 Paper D

Scheduling Real-Time Packets with Non-Preemptive Regions on Priority-based NoCs, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte.

In Proceedings of the 22th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), 2016.

In this paper, we present a new scheduling framework of wormhole-switched NoCs with virtual-channels. We introduce non-preemptive regions to NoC packets, and we present how to select a suitable length of such regions. A schedulability test of the proposed framework is provided. According to the evaluation results, the proposed approach can always achieve higher schedulability ratios compared to the original flit-level preemptive NoC. This paper addresses Research Contribution 2.

5.5 Paper E

A Tighter Recursive Calculus to Compute the Worst-Case Traversal Time of Real-Time Traffic over NoCs, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), 2017.

In this paper, we present an improved timing analysis (called TRC) to compute worst-case traversal delays of real-time packets over Round-Robin Arbitration (RRA) based wormhole-switched NoCs. According to the evaluation results, the proposed TRC can provide significantly tighter estimates compared to the original Recursive Calculus especially when the network contains a large number of flows. On the other hand, compared to the Branch and Prune/ Branch, Prune and Collapse (BP/BPC) algorithm, which can also provide tighter estimates, TRC requires much shorter computation time. This paper addresses Research Contribution 3.

5.6 Paper F

Buffer-Aware Analysis for Worst-Case Traversal Time of Real-Time Traffic over RRA-based NoCs, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), 2017.

In this paper, we identify an optimistic problem involved in the Recursive Calculus (RC) and other related timing analysis approaches. We propose a Revised Recursive Calculus (RRC) which extends RC by taking buffer-effects into account and supporting packetization of large packets. A case study based on an autonomous vehicle application is performed to evaluate the proposed analysis. The case study clearly shows the unsafe estimates produced by RC.

On the other hand, the estimates computed by RRC are always higher than the observations from the simulation. This paper addresses Research Contribution 3.

5.7 Paper G

Using Segmentation to Improve Schedulability of RRA-based NoCs with Mixed Traffic, Meng Liu, Matthias Becker, Moris Behnam, Thomas Nolte. In Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), 2017.

In this paper, we introduce a segmentation-based approach in order to improve the schedulability of real-time traffic in RRA-based NoCs. The proposed solution is also used to address the problem of transmitting both real-time traffic and best-effort traffic in the same NoC. The solution aims to provide low latency for best-effort traffic while the schedulability of all the real-time traffic is still guaranteed. According to the evaluation results, the proposed segmentation solution can significantly improve the schedulability of the whole network. This paper addresses Research Contribution 4.

Bibliography

- [1] W. J. Dally and C. L. Seitz, “Deadlock-free message routing in multiprocessor interconnection networks,” *IEEE Transactions on Computers*, vol. C-36, no. 5, pp. 547–553, 1987.
- [2] L. M. Ni and P. K. McKinley, “A survey of wormhole routing techniques in direct networks,” *Computer*, vol. 26, no. 2, pp. 62–76, 1993.
- [3] M. Shaw, “The coming-of-age of software architecture research,” in *23rd International Conference on Software Engineering (ICSE)*. IEEE Computer Society, 2001, pp. 656–664.
- [4] F. P. Brooks, “Grasping reality through illusioninteractive graphics serving science,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 1988, pp. 1–11.
- [5] É. Cota, A. de Morais Amory, and M. S. Lubaszewski, *Reliability, Availability and Serviceability of Networks-on-chip*. Springer Science & Business Media, 2011.
- [6] Z. Shi and A. Burns, “Real-time communication analysis for on-chip networks with wormhole switching,” in *2nd IEEE/ACM International Symposium on Networks on Chip (NOCS)*. IEEE, 2008, pp. 161–170.
- [7] B. Nikolić and S. M. Petters, “EDF as an arbitration policy for wormhole-switched priority-preemptive NoCs: Myth or fact?” in *14th International Conference on Embedded Software (EMSOFT)*. ACM, 2014, pp. 28:1–28:10.
- [8] L. S. Indrusiak, “End-to-end schedulability tests for multiprocessor embedded systems based on networks-on-chip with priority-preemptive ar-

- bitration,” *Journal of systems architecture*, vol. 60, no. 7, pp. 553–561, 2014.
- [9] D. Wentzlaff, P. Griffin, H. Hoffmann, L. Bao, B. Edwards, C. Ramey, M. Mattina, C.-C. Miao, J. F. Brown III, and A. Agarwal, “On-chip interconnection architecture of the tile processor,” *IEEE Micro*, pp. 15–31, 2007.
- [10] *Epiphany Architecture Reference*, Adapteva Inc., 2012.
- [11] K. Goossens, J. Dielissen, and A. Radulescu, “Æthereal network on chip: concepts, architectures, and implementations,” *IEEE Design & Test of Computers*, vol. 22, no. 5, pp. 414–421, 2005.
- [12] C. Paukovits and H. Kopetz, “Concepts of switching in the time-triggered network-on-chip,” in *14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. IEEE, 2008, pp. 120–129.
- [13] M. Schoeberl, F. Brandner, J. Sparsø, and E. Kasapaki, “A statically scheduled time-division-multiplexed network-on-chip for real-time systems,” in *6th IEEE/ACM International Symposium on Networks on Chip (NOCS)*. IEEE, 2012, pp. 152–160.
- [14] W. Dally, “Virtual-channel flow control,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 3, no. 2, pp. 194–205, 1992.
- [15] Z. Shi and A. Burns, “Real-time communication analysis with a priority share policy in on-chip networks,” in *21st Euromicro Conference on Real-Time Systems (ECRTS)*. IEEE, 2009, pp. 3–12.
- [16] —, “Schedulability analysis and task mapping for real-time on-chip communication,” *Real-Time Systems*, vol. 46, no. 3, pp. 360–385, 2010.
- [17] B. Nikolić, H. I. Ali, S. M. Petters, and L. M. Pinho, “Are virtual channels the bottleneck of priority-aware wormhole-switched NoC-based many-cores?” in *21st International Conference on Real-Time Networks and Systems (RTNS)*. ACM, 2013, pp. 13–22.
- [18] H. Kashif, S. Gholamian, and H. Patel, “Sla: A stage-level latency analysis for real-time communication in a pipelined resource model,” *IEEE Transactions on Computers*, vol. 64, no. 4, pp. 1177–1190, 2015.

- [19] Q. Xiong, Z. Lu, F. Wu, and C. Xie, “Real-time analysis for wormhole noc: Revisited and revised,” in *26th edition on Great Lakes Symposium on VLSI (GLSVLSI)*. ACM, 2016, pp. 75–80.
- [20] T. Ferrandiz, F. Frances, and C. Fraboul, “A method of computation for worst-case delay analysis on spacewire networks,” in *4th International Symposium on Industrial Embedded Systems (SIES)*. IEEE, 2009, pp. 19–27.
- [21] D. Dasari, B. Nikolić, V. Nélis, and S. M. Petters, “Noc contention analysis using a branch-and-prune algorithm,” *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 13, no. 3s, p. 113, 2014.
- [22] L. Abdallah, M. Jan, J. Ermont, and C. Fraboul, “Wormhole networks properties and their use for optimizing worst case delay analysis of many-cores,” in *10th IEEE International Symposium on Industrial Embedded Systems (SIES)*. IEEE, 2015, pp. 1–10.
- [23] E. A. Rambo and R. Ernst, “Worst-case communication time analysis of networks-on-chip with shared virtual channels,” in *18th Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2015, pp. 537–542.
- [24] L. Abdallah, M. Jan, J. Ermont, and C. Fraboul, “Reducing the contention experienced by real-time core-to-i/o flows over a tilera-like network on chip,” in *28th Euromicro Conference on Real-Time Systems (ECRTS)*, 2016, pp. 86–96.
- [25] B. de Dinechin, D. van Amstel, M. Poulhies, and G. Lager, “Time-critical computing on a single-chip massively parallel processor,” in *17th Design, Automation and Test in Europe Conference and Exhibition (DATE)*. IEEE, 2014, pp. 1–6.
- [26] W. Puffitsch, R. B. Sørensen, and M. Schoeberl, “Time-division multiplexing vs network calculus: A comparison,” in *23rd International Conference on Real Time and Networks Systems (RTNS)*. ACM, 2015, pp. 289–296.
- [27] Z. Shi and A. Burns, “Improvement of schedulability analysis with a priority share policy in on-chip networks,” in *17th International Conference on Real-Time and Network Systems (RTNS)*, 2009, pp. 75–84.

- [28] M. Joseph and P. Pandya, "Finding response times in a real-time system," *The Computer Journal*, vol. 29, no. 5, pp. 390–395, 1986.
- [29] H. Ayed, J. Ermont, J. I. Scharbarg, and C. Fraboul, "Towards a unified approach for worst-case analysis of tilera-like and kalray-like noc architectures," in *12th IEEE World Conference on Factory Communication Systems (WFCS)*, 2016, pp. 1–4.
- [30] J.-Y. Le Boudec and P. Thiran, *Network calculus: a theory of deterministic queuing systems for the internet*. Springer Science & Business Media, 2001.
- [31] Y. Qian, Z. Lu, and W. Dou, "Analysis of worst-case delay bounds for on-chip packet-switching networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 5, pp. 802–815, 2010.
- [32] T. Ferrandiz, F. Frances, and C. Fraboul, "A network calculus model for spacewire networks," in *17th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, vol. 1. IEEE, 2011, pp. 295–299.
- [33] R. Henia, A. Hamann, M. Jersak, R. Racu, K. Richter, and R. Ernst, "System level performance analysis—the symta/s approach," *IEE Proceedings-Computers and Digital Techniques*, vol. 152, no. 2, pp. 148–166, 2005.
- [34] A. E. Kiasari, Z. Lu, and A. Jantsch, "An analytical latency model for networks-on-chip," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 1, pp. 113–123, 2013.
- [35] M. Lai, L. Gao, N. Xiao, and Z. Wang, "An accurate and efficient performance analysis approach based on queuing model for network on chip," in *28th International Conference on Computer-Aided Design (ICCAD)*. ACM, 2009, pp. 563–570.
- [36] Z. Qian, D.-C. Juan, P. Bogdan, C.-Y. Tsui, D. Marculescu, and R. Marculescu, "Svr-noc: A performance analysis tool for network-on-chips using learning-based support vector regression model," in *16th Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2013, pp. 354–357.

- [37] M. Liu, M. Becker, M. Behnam, and T. Nolte, "Buffer-aware analysis for worst-case traversal time of real-time traffic over rra-based nocs," in *25th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, 2017.
- [38] J. L. Díaz, D. F. García, K. Kim, C.-G. Lee, L. L. Bello, J. M. López, S. L. Min, and O. Mirabella, "Stochastic analysis of periodic real-time systems," in *23rd IEEE Real-Time Systems Symposium (RTSS)*. IEEE, 2002, pp. 289–300.
- [39] M. Liu, M. Behnam, and T. Nolte, "A stochastic response time analysis for communications in on-chip networks," in *21st International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. IEEE, 2015, pp. 237–246.