RESEARCH STATEMENT
Deepak Gangadharan (deepakg@seas.upenn.edu)

The rise of connectivity among different types of devices/entities in the cyber-physical world has led to a technological revolution facilitating the creation of interconnected smart and intelligent systems. The advent of such systems has enabled efficient delivery of different kinds of services in areas such as transportation, smart homes, smart grid, medical and healthcare, etc. Such a large conglomeration of connected devices/entities form the Internet of Things (IoT). The large scale integration of heterogenous entities in an IoT brings with it the inherent complexity of system level design and analysis. In addition, these IoT systems are comprised of a large number of real-time embedded systems, which require schedulability analysis and design techniques specific to real-time systems. My primary research interests address these challenges and fall under the broad theme of “Scalable Design and Performance Analysis of IoT and Real-Time Embedded Systems”, especially delving into topics like edge computing based services for connected vehicles and/or safety critical real-time embedded computing systems for automotives.

One of the major challenges in performance analysis of applications running in IoT and Real-Time Embedded (RTE) systems is the predictability of performance objectives. Predictability in RTE systems is influenced by the variability in available computation, communication and memory resources for processing an application as a result of contention on the resources from competing tasks. However, the RTE system is typically confined in space and dimension and its performance analysis is not influenced by network communication technology. On the other hand, the performance analysis of an IoT system needs to consider a larger system view involving network communication technology and possibly user input devices in addition to the RTE sub-systems. There are common performance objectives considered in both an IoT system and a RTE system such as end-to-end delay, energy consumption, etc. However, the factors affecting predictability of these performance objectives in IoT systems are many more in comparison to the RTE systems. This poses the requirement for developing robust and efficient analysis methods taking into consideration specific contexts of the two systems in order to analyze the bounds on the performance objectives.

Optimal design of IoT and RTE systems is a very complex task due to the large input problem sizes encountered. For example, in an automotive IoT, as the number of vehicles increases, the task of determining an optimal system configuration becomes very complex. Similarly, with increase in the number of tasks in a RTE system, it becomes more complex to determine what real time parameters result in the optimal performance. Therefore, I would also like to propose and develop novel algorithms to design optimal IoT and RTE systems while being scalable to increasing problem size, i.e., efficiently determine their system parameters for large real-life problems that result in optimal performance. In general, I envision to work towards proposing novel design and analysis techniques for timing predictability, energy/thermal efficiency, system resilience and optimal resource usage.

In my previous works, I have developed formal analysis techniques to perform worst/average-case analysis in various problem contexts pertaining to executing hard/soft real-time applications on multiprocessor system-on-chips (MPSoCs). Most of the techniques developed had its theoretical roots in Real-Time Calculus (RTC) (a well known tool for compositional analysis of distributed systems) and partly also in resource reservation-based scheduling. Further, I have also developed analytical techniques based on Stochastic Network Calculus to derive quality-of-service (QoS) quantified as probabilistic bounds on performance for multimedia applications running on MPSoC platforms.

1. Performance Analysis and Design of RTE systems

Firstly, I will present the prior work I have accomplished in the area of performance analysis and design of RTE systems and will subsequently throw light on some interesting problems that I am looking forward to solve in future.


As a PhD student, I extensively worked on developing new performance analysis techniques for multimedia MPSoC platforms. Due to the inherent burstiness and workload variability of multimedia data, it is non-trivial to compute the worst-case delay that an incoming stream suffers or the maximum backlog that would be incurred in the platform. This variability in multimedia data is very well characterized using interval-based curves from RTC, which are used to characterize both the application workload demand and allocated processor service. These workload and service curves are then used to analyze the non-functional system properties.

In my thesis, I mainly focused on resource allocation problems in MPSoC platforms with soft real-time scenarios. These problem scenarios required analysis approaches that needed to consider some data losses at the buffer resources. Con-
vitional RTC operations were based on the notion of infinite buffer size at the input/output of a processing resource. However, in order to accommodate bounded data losses with objective quantification of quality degradations incurred, I proposed a quality-aware extension to the RTC framework. This new analysis framework considered finite buffer sizes and enabled a quality-aware performance analysis of various non-functional objectives.

Based on the coupling of quality parameter (i.e., peak signal to noise ratio (PSNR)) of a multimedia stream into the RTC framework for finite buffer sizes, I proposed three novel analysis techniques to determine buffer size, processing bandwidth and thermal capacity in order to decode encoded video streams on MPSoC platforms under quality constraints.

1. Firstly, a mathematical framework was proposed [1], which derives the buffer sizes required before and after every processing unit or server such that a pre-specified quality constraint given by the system designer is satisfied under an available processing service. This analysis can also be used to trade-off buffer sizes with the quality values inside a design space exploration framework for deriving a multimedia MPSoC architecture. In the same line of work, I also proposed a quality driven buffer dimensioning algorithm [2] where I use the motion vector parameter from the encoded video stream to determine the priority of frames dropped while adhering to a target quality constraint.

2. The second analysis problem [3] proposed a solution to derive the minimal processor bandwidth requirements for multiple media streams sharing a processor such that each stream has a target quality constraint to be satisfied. This problem is not an inverse of the first problem (which would lead to very pessimistic results), but is more complicated due to sharing of the processor bandwidth. It was then shown how the derived bounds on processor service for two streams can be used to derive the parameters of a TDMA scheduler.

3. Finally, an approach [4] that combines an application level technique (namely frame drops) with dynamic thermal management (DTM) policy was proposed to process multimedia streams (video frames in this context) satisfying both quality as well as thermal constraints. It is a combined offline and online method where some stream information generated offline was used to optimize the idle time introduction online. A lightweight history of processing requirements of a limited number of elementary units of the media stream was maintained to further achieve reductions in the idle times introduced.

To summarize, my PhD thesis proposed several important techniques to mathematically analyze and derive appropriate multimedia MPSoC non-functional parameters in a quality-aware manner. In addition, I proposed techniques to perform fast hybrid simulation [5, 6] for performance analysis of multimedia MPSoC platforms. As part of the broad research focus on quality-aware performance analysis for multimedia MPSoC platforms, I also proposed two probabilistic analysis approaches for multimedia MPSoC platforms. In the first problem [7], the frequency allocated to a processing resource could vary randomly while the incoming multimedia stream rate and their execution requirements were considered to be deterministic. An optimal playout delay value was derived such that the probability of underflow in playout buffer was upper bounded to a desired value. In the second problem [8] scenario, randomness was considered in the incoming multimedia stream objects. An analysis approach based on Stochastic Network Calculus was proposed to derive probabilistic bounds on the buffer underflow.

1.2. Analysis and Design Space Exploration for Mapping Applications on MPSoC Platforms

As a PostDoctoral Researcher, I have worked on several problems delving into proposing new analysis techniques for better timing predictability of concurrent applications running on MPSoC platforms. The direction of the first work [9] was to explore the concepts of composability in hybrid design space exploration (DSE) of NoC-based multi-tile platforms using spatial isolation on shared resources. In this work, we achieved composable mapping of applications by first deriving spatially isolated feasible mapping solutions offline and then choosing the appropriate feasible solution at run time with the help of a backtracking algorithm.

Subsequently, I also proposed a design-time/run-time application mapping methodology on MPSoCs [10], which employs temporal isolation in order to ensure timing predictability of an application mix. In this work, we proposed a novel constraint satisfaction approach to choose appropriate feasible solution at run time while incorporating temporal isolation. This method gave us better resource utilization results compared to the approach where only spatial isolation was used. This journal publication has been submitted after minor revision and is expected to be accepted soon.

In large heterogeneous MPSoC platforms, accessing shared resources such as cache/memory and buses by multiple processors may lead to significant contention on them, thereby affecting the timing predictability adversely. I have worked on problems that address the issue of contention on buses by firstly developing a method to reconfigure the AMBA AHB bus at runtime [11] and then proposing an online algorithm to perform reconfiguration of bus scheduling policy between non-preemptive fixed priority (NPFP) scheduling to time division multiple access (TDMA) scheduling with low complexity [12]. This work demonstrated how effectively the bus scheduling policy can be chosen on the fly to service as many
I have also proposed DSE and synthesis techniques for platforms consisting of multiple application-specific instruction set processors (ASIPs) [13, 14]. The underlying idea in these works was to use a probabilistic DSE approach to derive solutions which provide the mapping of an application on a multi-ASIP architecture. In this work, I proposed a novel probabilistic DSE based solution to break the cyclic dependency (determination of ASIP architecture dependant on task clustering and WCET used to determine schedulability dependant on ASIP architecture synthesis) by proposing a WCET uncertainty model which is used in a stochastic schedulability analysis approach encapsulated within a genetic algorithm. This work was conducted considering a periodic application scenario [13] and a event triggered application scenario [14].

1.3. Future Research Directions

Given the extensive work I have been doing in RTE systems, my future research will also be focusing on such systems. Under this theme, I will be working on problems in scheduling and analysis of RTE systems delving into aspects such as data freshness, security over shared resources, mixed criticality and multimode system operation. I plan to work on these aspects while ensuring timeliness, energy/thermal efficiency and resilience objectives. All these themes will be primarily explored in the context of RTE systems relevant to automotive industry.

1.3.1. Data freshness aware uniprocessor/multiprocessor scheduling

The notion of data freshness is important in the correct execution of a safety critical application in automobiles. This is because the freshness of data from a sensor to be used by safety critical tasks needs to be within allowed freshness bounds in order to satisfy the safe operation of the system. There may also be freshness requirements all the way down the chain of tasks. At the early stage of RTE system design, when not much information about the underlying platform is available, it is difficult to determine the real time parameters of tasks for execution while satisfying the freshness constraints. One of my recent papers [15] (won the best paper award), which was co-authored with a graduate student I mentored, proposed a method to determine the periods of tasks in chains of arbitrary length while satisfying end-to-end freshness constraints with only few assumptions regarding the scheduling algorithm used.

I plan to continue this line of research by proposing a data freshness aware scheduling technique for energy/temperature minimization. In the previous work [15], it has been found that the lower the assigned periods of the producer tasks in a chain, smaller is the achieved end-to-end freshness bound for the task chain. However, lower the period of the tasks, higher is the necessary frequency of operation and lesser is the opportunity to introduce slack time/sleep time in the schedule and therefore higher is the energy consumption/peak temperature of the RTE system. Therefore, the potential problem is to design a RTE system by determining the periods of tasks that explore the trade-offs between the conflicting objectives of data freshness and energy consumption/peak temperature.

Current RTE systems are susceptible to faulty execution due to external attacks or faults in the underlying platform. Such a behaviour in the RTE system can adversely affect the freshness bound of task chains. Therefore I would like to explore the effect of faults on data freshness and formulate conditions that will minimize the effect of faults on data freshness. Alternatively, I plan to propose a resilient scheduling technique and its corresponding analysis method to determine the end-to-end freshness bound for a given number of faults in the system.

1.3.2. Security-aware uniprocessor/multiprocessor dynamic scheduling

Security is a critical requirement when multiple tasks use a shared resource for their execution in uniprocessor or multiprocessor systems. For instance, when a low security task is scheduled to execute after a high security task, the information of the high security task on the shared resource like cache becomes potentially vulnerable for leakage. In order to avoid this information leakage, flushing of the shared resource before the execution of the low security task is one of the recommended techniques [16]. The additional time incurred for flushing is an overhead to the scheduling algorithm used. There has been considerable work done to perform analysis of what is the worst-case number of flushes required and what is the corresponding result on the worst-case response time (WCRT) of tasks under fixed priority scheduling algorithms. However, there is no work in scheduling tasks in a security-aware context with dynamic priority scheduling algorithms like EDF while meeting the constraints of tasks requiring high security for their data.

The security-aware scheduling work has promising research directions under the context of multi-mode systems. A multi-mode system is very relevant in the context of an automotive RTE system where the vehicle passes through different modes during its trip based on external and internal triggers. Given a set of modes that a system passes through, the problem to derive the worst-case number of flushes is even more complex than the earlier scenario of single mode system. This is because the changes in the task mix that dynamically occur as a result of mode change directly affects the number
of flushes required. Additionally, for a worst-case analysis, the important challenge that the analysis technique needs to
tackle is to determine what are the mode transition points that lead to worst-case number of flushes.

### 1.3.3. Scheduling and Analysis for Mixed Criticality and Multi-Mode RTE systems

Scheduling and analysis of mixed-criticality task chains (involving dependencies) on multiprocessor systems has receive
very little attention in literature. One important problem that requires attention in this area is how to efficiently and
precisely derive WCRT of tasks under mixed critical scheduling algorithms like EDF-VD. The main challenges here are
to determine how much of the workload carried over during criticality change affects the WCRT of high criticality tasks
and developing a method to precisely compute the WCRT. I have extensively studied this problem and have proposed
a solution to derive the WCRT of mixed critical tasks under EDF-VD with no task dependencies. This work is under
preparation for a publication. However, precisely computing the end-to-end latency of a task chain (with dependencies)
consisting of mixed critical tasks under partitioned EDF-VD on multiprocessor systems is still an open problem.

There is very little work in the area of analysis and design of mixed criticality multi-mode (MCMM) RTE systems, which
is very relevant to automotive platforms. The task of precisely computing the end-to-end latency of a task chain for
MCMM RTE systems is very complex because apart from the criticality mode change, the RTE system also experiences
mode changes at the system level. These mode transitions can happen independently, which makes the problem even
more challenging. The challenge here is to develop a scalable algorithm to compute the worst-case end-to-end latency
of a task chain by taking into consideration all possible combinations of criticality and system mode switches. I would
like to use the results that I will obtain from the earlier proposed work on mixed criticality to derive precise estimates of how
much workload spills over during system mode transitions. The worst-case workload carried over across criticality mode
switches can then be used in conjunction with the system mode switches to compute the worst-case end-to-end latency.

### 2. Analysis and Design of Edge Computing-Based Automotive IoT systems

The impetus towards increased vehicle connectivity in the automotive industry is mainly driven by the requirement to
provide more safety, efficiency and better user experience. Connectivity has enabled the delivery of useful data/services to
the vehicles on the move. It can be delivered from the cloud server directly or via the edge nodes (which are located closer
to the vehicles). Delivery of data/services to vehicles via the edge while the vehicle is on the move requires allocation
of adequate amount of memory, computation and communication resources on the edge nodes so that a certain system
objective is satisfied.

In my previous work [17], I formulated this data/service delivery problem as an optimization problem, which minimizes
the system wide total bandwidth cost of the edge nodes. I also studied the effect of variation of different traffic flow model
parameters on the bandwidth cost. As a follow up work, I proposed a multi-objective optimization framework [18] for
data/service delivery considering both delivery time and total edge bandwidth cost. However, the optimization frameworks
only dealt with the scenario where the vehicles followed a fixed pre-specified route as in the case of a delivery truck fleet
belonging to a company, where the routes are assigned beforehand to the truck drivers. If a driver changes the route
taken based on route/traffic conditions, then our previous work will not be applicable. The proposed optimization also
increases in time complexity with increase in the number of edges and vehicles.

#### 2.1. Future Research Directions

The lines of research that I propose to pursue in future within the area of edge computing-based automotive IoT system
broadly fall under the following themes

##### 2.1.1. Handling dynamic route changes

In the earlier work, it was assumed that the vehicles need to publish their routes to the cloud and follow their published
route to destination. However, this assumption does not hold if the driver decides to drift away from the published
route for some reason not known a priori. One solution that I would like to explore in this context is to determine the
feasibility of a runtime algorithm that will make decisions to transfer the data/service to the appropriate edges when
the vehicle changes the route. This algorithm will make online decisions about what is(are) the best edge(s) to forward
the data/service to based on the results from the centralized optimization in the cloud and the information that edges
exchange between themselves.

##### 2.1.2. Addressing scalability issues

One of the main problems with the current solution for data/service delivery is its scalability. An increase in the number
of edges and vehicles will push the optimization runtime into tens of minutes or hours and it may end up being prohibitive
for a real-time dispatch of data/services to the edges. I plan to address this problem using two approaches. The first
approach will be to develop a low complexity heuristic algorithm, which can be run online and will deliver data/service
to the vehicles such that the solution is close to optimal. In the second approach, I envisage to develop a partitioned
optimization approach where the optimization will be performed on a smaller area consisting of lesser number of edges
and vehicles. The results from the local optimizations (which will be comparatively faster) will then be composed at the
global level to get solutions close to optimal for the whole system.

2.1.3. Prototype implementation of data/service delivery for safety critical applications
In my opinion, the full utility of a data/service delivery framework is realized only when it is made practical for safety
critical applications such as ADAS applications. Under this theme, I envisage to undertake an ambitious plan to imple-
ment a prototype version of a hierarchical connected vehicle architecture consisting of the cloud, edge devices and multiple
mobile units. I also plan to implement state-of-the-art communication protocols for data transfer between cloud, edges
and mobile units. This testbed will then be used to validate algorithms that distribute data/service for safety critical
automotive applications.

2.1.4. Performance analysis of edge computing for deep learning algorithms
In addition to data/service delivery for connected vehicles, one other area that I have started exploring is the use of deep
learning (DL) algorithms for connected vehicle applications using the edge device. Vehicle Platoon Control is one such
connected vehicle application which can greatly benefit from DL based edge analytics whereby vehicles can learn a shared
prediction model of the environment in which they are moving. This can reduce the amount of data generated by sharing
mutual information and as a result reducing the burden on communication and computation. As DL based edge analytics
is performed on resource constrained edge devices, I see the potential to conduct research on DL algorithms that use
intelligent information generation and sharing mechanisms to optimize the energy efficiency and thermal requirements of
the edge device.

2.1.5. Performance analysis of concurrently executing safety critical automotive task and deep learning
based inference task on a edge device
Currently, there is a lot of research on developing efficient DL algorithms to execute on the resource constrained edge
devices. One direction in this research is to trade off accuracy of the DL inference by pruning some feature data and
thereby reducing resource usage and time. If there are safety critical automotive tasks that need to be executed on the
edge device (for e.g., offloaded tasks from vehicles) along with the DL inference task, a pertinent problem will be to analyze
how much accuracy of the DL task needs to be traded off with the timeliness of the safety critical tasks, in the worst-case,
given an arrival pattern of the safety critical tasks. This problem can also be explored by considering computing on more
than one edge device. The relevant question to answer in this case would be to determine the minimal number of edge
nodes required to execute the DL inference and safety critical tasks such that a target inference accuracy and timeliness
properties are respectively satisfied.

References
RTCSA, 2011, pp. 319-328.
pp. 976-981.
ASAP, 2014, pp. 256-263.
multimedia MPSoC platforms”, CODES+ISSS, 2009, pp. 413-422.
in resource-constrained media players”, CODES+ISSS, 2011, pp. 169-178
modeling and performance analysis of multimedia SoCs”, SAMOS, 2013, pp. 145-154


D. Gangadharan, O. Sokolsky, I. Lee, B. Kim, C.-W. Lin, S. Shiraishi, “Bandwidth Optimal Data/Service Delivery for Connected Vehicles via Edges”, Accepted to IEEE CLOUD, 2018 (Selected as one of the best papers from IEEE CLOUD and invited for a journal publication).