Master Thesis Proposals

Contact:
Federico Reghenzani
federico.reghenzani@polimi.it

Last update: June 2024

This file contains a general description of the topics for a possible thesis. The research environment is very dynamic, and therefore the thesis topics. This document describes more the research lines than the actual theses: feel free to contact me for further details.

Prerequisites

Enthusiasm and goodwill.

It is not expected to have in-depth knowledge of the following topics by the student, and the common background of Master's Students at Politecnico is sufficient.

Area 1 – Real-Time Systems

1.1 - Real-time scheduling and WCET analyses

Traditional analyses of the Worst-Case Execution Time (WCET) failed to follow the technology trend of the modern processor architectures. This makes the use of modern computers difficult in hard real-time scenarios. Systems in this category are usually required to prove their compliance with timing requirements due to safety-criticality aspects. Some examples include aerospace applications, medical devices, and automotive systems. Novel approaches need to be studied.

Possible master theses include theoretical and experimental works. Several aspects can be the subject of a thesis, for instance:

- Design of a custom processor architecture
- Software-level design (including operating systems, schedulers)
- Timing measurements and statistical techniques
- Use of GPUs and FPGAs for real-time workload

1.2 - Mixed-Criticality Systems

System consolidation, a traditional trend in server and cloud fields, has recently appeared also in the embedded world in order to reduce design and maintenance costs. Typical examples are automotive and avionics applications. However, different applications may have different priorities and requirements, leading to the necessity of complex scheduling decisions when they run on the same platform. Systems, where different applications with different criticality levels co-exist, are called mixed-criticality systems. For example, an automotive computer can run both the *electronic stability control* program (hard real-time) and the applications for the infotainment (soft real-time).

Managing such systems is not easy and presents several intriguing challenges. Possible thesis topics are related to:

- Operating system design
- Design of mixed-criticality applications with real use cases (drones, automotive, etc.)
- Non-functional optimization (energy, reliability, etc.)

1.3 Compilers and real-time

Modern compilers perform many operations “under the hood” that the programmer often ignores. The compiler optimizations commonly target better average-case metrics, however, in real-time ew are interested more in the worst-case. Can we modify a modern compiler (LLVM) to better fit the real-time requirements? How can we exploit the intrinsic knowledge that the compiler obtains while performing code transformations?
Possible examples:

- Compiler optimization tailored for WCET minimization
- Compiler-provided information to scheduler
- Timing analysis of the compiler optimizations

Area 2 – Fault Tolerance

The problem of being resilient to faults is becoming a concerning problem for several classes of computer systems because of the continuous miniaturization of the components, which makes them more susceptible to radiation and other environmental conditions. The problem is exacerbated when the embedded system is running in extreme environments such as aerospace applications, or when the system is composed of thousands of computing nodes (High Performance Computing). Software fault-tolerant mechanisms are well-known and already available in scientific literature. They react to faults (detected by the hardware or by the software) by re-executing the workload, running recovery routines, or other techniques to allow the system to produce a correct result.

2.1 Software fault-tolerance

While many state-of-the-art techniques exist, the advance of the technology, particularly the introduction of multi-/many-core, makes the development of novel approaches possible. Software fault-tolerance techniques must also consider their added overhead in the system, especially important for real-time systems. Exploring novel techniques, with the help of recent hardware features, aiming at dealing with the trade-off of fault-tolerance requirements and other requirements (energy, power, etc.) raises new challenges. Possible master's theses include:

- Development of novel techniques and comparison with state-of-the-art techniques
- Analysis of existing techniques, their limitations, and possible solutions at OS-level
- Studying how to apply fault-tolerance techniques to system software (e.g., scheduler, ...)

2.2 Fault-tolerance and real-time systems

Introducing fault-tolerance techniques in a real-time environment makes the scheduling problem difficult because of the introduction of extra tasks not performing useful work when no fault occurs. This situation has two unwanted consequences: it makes the satisfaction of real-time requirements difficult and, due to preemption, may expose the 'normal' tasks for a larger period of time to transient faults. This is a very active area in research and presents numerous challenges. Possible thesis topics include:

- Scheduling and resource management strategies for real-time systems
- Analyses of the reliability and timing requirements and their interactions
- Study of specialized architectures and use-case implementation (e.g., dual-socket real-time + general-purpose processors)

2.3 Software fault-detection

Novel techniques to detect faults at the program level are essential to moving fault tolerance from hardware devices to software routines. We would like to investigate a possible solution to exploit timing information to detect behavioral changes of the applications and how to distinguish them from faults. Possible topics:

- Application characterizations and online verification of the behaviors (anomaly detection)
- Offline and online probabilistic analyses on the execution time

FAQ

Can I propose a topic for the thesis? Because I participate in a project / I have an idea / I'm working on something / ...
Yes, but be aware that it should somehow be related to my research topics. Feel free to contact me so that we can discuss if it is possible to create a thesis from your idea/project.

I am a first-year master student, should I wait until the second year to ask the thesis?
No, it is not necessary. However, the timeline of your thesis should be tuned accordingly and some proposals, which have strict deadlines, may not be available.

Should I wait to have completed all the exams to start with the thesis?
No, it is not necessary, and I discourage it. There are some preliminary activities that can be done in parallel with your exams, and you can save a lot of time if you do them in parallel.

I am interested in the topics but I am not a Computer Science student, can I ask for a thesis?
It depends on the answers of these two questions: 1) Is the topic you have in mind cross-disciplinary among my research and your study course? 2) Does your study course allow a professor of 09/H1 SSD to be your advisor? (check the rules of your school)

Other ideas?

Feel free to contact me if you have any proposal on your side and we will discuss if it is possible to create a thesis from it. Be aware that it should somehow be related to my research topics.