



# RT-SPACE: A Real-Time Stochastically-Provisioned Adaptive Container Environment

Department of Computer Science

University of North Carolina at Chapel Hill

August 2014

## The Challenge

Many envisioned forward-looking cyber-physical systems, particularly in military domains, will have timing constraints that are difficult to ensure using current real-time design and analysis techniques. For example, future autonomous unmanned air vehicles (UAVs) must be capable of reacting to (potentially many) threats within predictable time durations. Similarly, an autonomous ground vehicle must be able to detect obstacles, including potentially lethal ones such as improvised explosive devices, and react accordingly. In these settings, significant computational processing power is needed (e.g., so that more threats can be handled) and the temporal requirements of the supported workload may change dynamically at runtime (e.g., as new threats are detected).

These requirements suggest the usage of multicore platforms, which can be leveraged to provide the necessary processing power while realizing size, weight, and power (SWaP) benefits. Unfortunately, the parallelism inherent in such platforms has long been seen as a stumbling block when addressing real-time-related resource allocation problems. Such problems can be particularly difficult to deal with if all timing constraints are expressed as **hard real-time (HRT)** per-task deadlines that can never be missed. HRT systems typically require conservative resource allocation and pessimistic analysis for checking timing constraints that can cause significant unutilized or “lost” processing capacity. In a multicore setting, this lost capacity can easily negate the extra processing capacity of any additional cores.

## The Approach

HRT constraints may be overkill in highly dynamic real-time applications such as those mentioned above. The very nature of these applications precludes certifying with certainty that timing constraints are never violated. For example, a UAV may be presented with so many threats that it is simply not tenable to handle them all. Thus, some degree of *uncertainty* is fundamental. In this case, the capacity loss associated with HRT approaches is actually harmful, as only workloads of lesser capacity can be certified. Is it better for a fielded system to be able to simultaneously handle ten threats successfully with high probability or only two threats with certainty? In

pondering this question, keep in mind that any claims of HRT “certainty” are based on mathematical abstractions that only approximate the physical system.

Motivated by these observations, we are developing real-time resource allocation infrastructure in the form of **RT-SPACE**, a **Real-Time Stochastically-Provisioned Adaptive Container Environment** for multicore-based applications. A “container” is a scheduling abstraction that allows a sub-system of tasks to be “contained” and isolated from the rest of the system. Such isolation facilitates the separate development and analysis of separate sub-systems. RT-SPACE will enable different application components to be temporally isolated and to be provisioned based upon average-case or near-average-case task execution times, instead of more pessimistic worst-case execution times. It will also enable such provisionings to be dynamically changed at runtime. The term “environment” connotes that RT-SPACE will be composed not only of methods for scheduling CPU time but also associated analysis for validating **soft real-time (SRT)** correctness. The development of such scheduling methods and analysis results will be the main intellectual contribution of this project. Additionally, an open source implementation of RT-SPACE will be produced based on LITMUS<sup>RT</sup>, a Linux-based real-time operating system originally developed at UNC under prior support from the U.S. Army Research Office.

## Significance

There is currently great interest in autonomous and robotic military systems with enhanced situational awareness that must be capable of analyzing complex situations and reacting accordingly in real time. The computational demands of such systems can be met by utilizing multicore platforms, but only if needed multicore-ready real-time resource allocation infrastructure is available. RT-SPACE will provide such infrastructure for those application domains where SRT correctness suffices.

## Project Members

James Anderson, professor  
Sanjoy Baruah, professor

**Research Sponsor**

U.S. Army Research Office

**For More Information**

Dr. James Anderson

Department of Computer Science

University of North Carolina at Chapel Hill

CB#3175, Sitterson Hall

Chapel Hill, NC 27599-3175

Phone: (919) 590-6057

Fax: (919) 590-6105

Email: anderson@cs.unc.edu

<http://www.cs.unc.edu/~anderson/projects/rtspace.html>