A Memorable Model with Great Guarantees

How important is it to be in full control? Where life coaches talk about controlling your decisions and health, others might speak about controlling conversations and politics. In this article we will talk about something far more predictable where many might think that control is given, and this is the control of electronics and machines.

The precision of our modern devices can reach high standards and be precise down to the nanometer, but some real issues can occur when there is requirements on the timing. This is especially true in complex systems where many applications might have to work next to each other and share their resources. All applications wants to work, but they can't all run at the same time. Which one should be allowed first, and will this cripple the system system?

This unknown waiting times can be unacceptable in some situations, for example when it comes to safety. If you hit the brakes on a car, imagine if there wouldn't be a guarantee that the wheels will stop, or that the control system in a airplane wouldn't respond fast enough to the given directions. This is why predictable systems are of great importance.

Obviously this problem is solved in today's technology that's on the market, but systems are growing bigger by the year and are getting more and more complex. This is putting a pressure on the development of *scheduling algorithms*. The scheduling algorithm can be described as the doorkeeper that decides which process or application is allowed to run next. These need to be efficient, energy effective and make the right decisions but the processing power is reserved and the calculation speed is limited. However, we need guarantees for the schedulers that they will do what we expect.

The fact is that there are several scheduling algorithms available that are fast and effective, but they lack the guarantees for the complex systems. These scheduling algorithms tend to be optimal in specific situations. The situation could for example be a car where the control system must be very responsive for safety reasons and the media system is allowed longer delays. Another situation could be a media center with a lot of monitors that should show the same images on all screens, thus all applications needing an equal amount of power and image data.

So the question is, how can we create guarantees for these schedulers that are fast and effective? How can we be sure that they do what we want them to do before it is too late? One possible approach is the one made by rt-bench.

Rt-bench is an application that intend to perform experiments and find out how these schedulers behave in certain situations. It tries to create an environment as close to the reality as possible and measure the behaviour of a model application. The result of these tests can then be used to determine the behaviour of the application, supplying the guarantees that we so badly desire. For these guarantees to be valid however, we need to be sure that the conditions of our tests is realistic. The model needs to be as close to the real application as possible.

Rt-bench supply tools and building blocks to create a desired model, and these tools must be sufficient to be able to create a realistic model. This is why a memory handler has been added to the application. Close to all sophisticated systems have some sort of memory handling, where they save data or calculated values. This puts a certain kind of load on the systems and needs to be taken into consideration. By adding the memory handling, these systems can now be modeled with a higher sense of realism in rt-bench.

The anticipation for rt-bench is that all the fast and effective schedulers that engineers have been too afraid to adapt into their systems will now be able to prove themselves reliable. These reliable schedulers will find their way into our modern applications and create a better, faster and more effective world.