**Using Massively Parallel Processing in the Testing of the Robustness of Statistical Tests with Monte Carlo Simulation**

TAMÁS FERENCI *(Corvinus University of Budapest, Hungary, ft604@hszk.bme.hu)*
BALÁZS KOTOSZ *(Corvinus University of Budapest, Hungary, balazs.kotosz@uni-corvinus.hu)*

The *validity* of a statistical test is defined as its property of having equal Type I error rate and significance level. The *robustness* of a test is its property to be valid to some extent even if its underlying *assumptions* are not met for the sample(s).
One way to check these properties for a given test is to generate a large number of random samples for the test and perform testing. Samples would have same or varying relevant features depending on whether we are testing validity or robustness, respectively. Then, we compare the empirically found Type I error rate with the significance level: if they are "close enough", one can presume that the test was valid/robust. This is the so-called *Monte Carlo (MC)* method.
One common problem with MC methods is their extreme need for *computing performance*. Although personal computers are widely available today, and have an impressive computing capacity for everyday tasks, they are still unfit for larger scale MC simulations. Whilst common statistical packages like R or SPSS make MC methods possible with user-developed applications, such programs simply run too slow in complex situations (e.g. if multidimensional parameters specify the features of the generated samples) on PCs.
To overcome this limitation, we employed another approach (falling into the category of Massively Parallel Processing, or MPP), called GP (General Purpose) GPU-computing. It is based on the fact that the Graphical Processing Units (GPUs) found in modern video cards have a highly parallel architecture that makes the performing of a limited subset of algorithms (namely: well parallelizable algorithms, like MC) extremely fast. During this project we developed a program operating under NVIDIA's CUDA for the aim. (We tested the application with a middle-class video card.)
As an example, we examined the very typical question of the robustness of *Student's t-test* if its samples are coming from a non-normal distribution using this approach. (Samples were generated with Fleishman's polynomial power transformation method, with nonlinear equation-systems solved with GSL library.)
In our tests, we reached a peak performance of *214.5* million (!) random numbers generated (and *t*-tested with sample size=*10*) a second (with Mersenne Twister random number generator).
We tested the effect of skewness (in 20 steps from 0.0 to 4.0) and kurtosis (in 20 steps from 3 to 13). With 10 million sample-pairs generated at each step (sample size=*10*), this meant $20 \cdot 20 \cdot 10 \cdot 10000000 \cdot 2 = 80$ billion random number generations and 4 billion hypothesis testing – this was all done in 373 seconds!
The results showed that the test is highly sensitive for the departures from normality, but – more specifically – far more sensitive for skewness than for kurtosis. We also demonstrated the effects of the sample size (i.e. the effect of the central limit theorem).

The true power of our work lies in the fact that the testing environment we developed is flexible and scalable, so it can be easily adopted to virtually any statistical test – their properties can be tested with very high performance and without any programming (or GP-GPU) knowledge.

*Keywords*: Hypothesis tests, Monte Carlo simulation, validity, robustness, high performance computing