High-Energy Jet Interaction Monte Carlo for the Future Generations: HIJING++

Gábor Bíró[1], Gergely Gábor Barnaföldi[1], Gábor Papp[1,2], Péter Lévai[1], Miklós Gyulassy[1,3,4,5], Xin-nian Wang[3,4], Ben-wei Zhang[3]

[1]Wigner Research Centre for Physics of the Hungarian Academy of Sciences, Budapest, Hungary

[2]Institute for Physics, Eötvös Loránd University, Budapest, Hungary

[3]Institute of Particle Physics, Central China Normal University

[4]Nuclear Science Division, MS 70R0319, Lawrence Berkeley National Laboratory, Berkeley, California 94720 USA

[5]Pupin Lab MS-5202, Department of Physics, Columbia University, New York, NY 10027, USA

During the approaching second Long Shutdown of the Large Hadron Collider (LHC) in 2019-2020 many technical improvement will occur in the accelerator complex, in the detector systems and in the data acquisition systems. These will result in a huge increase of the number of expected collisions per second and as a consequence also the amount of measured data will grow rapidly. This period is the forerunner of the next generation of particle accelerators (HL-LHC, FCC), where we will accumulate experimental data in a higher rate than ever. In parallel we need to improve also the numerical tools in order to be able to keep up the requirements of the high-precision era.

The recently developed HIJING++ (Heavy Ion Jet Interaction Generator) will be the successor of the widely used original HIJING, developed almost three decades ago. While the old version was written in FORTRAN, the soon-to-be-published HIJING++ was completely rewritten in modern C+ +. During the development we keep in mind the requirements of the high-energy heavy-ion community: the new Monte Carlo software have a well designed modular framework, therefore any future improvement will be much easier. It contains all the improved physical models that were also present in it’s predecessor, but utilizing modern C++ features it also includes native thread based parallelism, an easy-to-use analysis interface and a modular plugin system, which makes room for possible future improvements like GPU acceleration. In my talk I present the current state of HIJING++ including the recent performance and benchmark tests.