LHC@Home: a BOINC-based volunteer computing infrastructure for physics studies at CERN

Javier Barranco[5], Yunhai Cai[7], David Cameron[4], Matthew Crouch[2], Riccardo De Maria[2], Laurence Field[2], Massimo Giovannozzi[2], Pascal Hermes[2], Nils Høimyr[2], Dobrin Kaltchev[8], Nikos Karastathis[2], Cinzia Luzzi[2], Ewen Maclean[2], Eric McIntosh[2], Alessio Mereghetti[2], James Molson[3], Yuri Nosochkov[7], Tatiana Pieloni[5], Ivan D. Reid[1], Lenny Rivkin[5], Ben Segal[2], Kyrre Sjobak[2], Peter Skands[6], Claudia Tambasco[5], Frederik Van der Veken[2], Igor Zacharov[5]

[1] Brunel University, London, Uxbridge UB8 3PH, UK
[2] CERN, 1211 Geneva Switzerland,
[3] LAL, Univ. Paris-Sud, CNRS/IN2P3, Université Paris-Saclay, Orsay, France
[4] University of Oslo, Postboks 1048 Blindern N-0316 Oslo Norway,
[5] Particle Accelerator Physics Laboratory, École polytechnique fédérale de Lausanne (EPFL)
1015 Lausanne Switzerland
[6] School of Physics & Astronomy, Monash University, VIC-3800, Australia,
[7] SLAC National Accelerator Laboratory, Menlo Park, CA, USA,
[8] TRIUMF, 4004 Wesbrook Mall, Vancouver, B.C., Canada V6T 2A3

Abstract

The LHC@Home BOINC project has provided computing capacity for numerical simulations to researchers at CERN since 2004, and has since 2011 been expanded with a wider range of applications. The traditional CERN accelerator physics simulation code SixTrack enjoys continuing volunteers support, and thanks to virtualisation a number of applications from the LHC experiment collaborations and particle theory groups have joined the consolidated LHC@Home BOINC project. This paper addresses the challenges related with traditional and virtualized applications in the BOINC environment, and how volunteer computing has been integrated into the overall computing strategy of the laboratory through the consolidated LHC@Home service. Thanks to the computing power provided by volunteers joining LHC@Home, numerous accelerator beam physics studies have been carried out, allowing for improving the understanding of charged particle dynamics in the CERN Large Hadron Collider (LHC) and its future upgrades. The main results are highlighted in this paper.

Keywords: volunteer computing, high energy physics, particle accelerators, beam dynamics

---

Copyright © by the paper's authors. Copying permitted for private and academic purposes.

In: E. Ivashko, A. Rumyantsev (eds.): Proceedings of the Third International Conference BOINC:FAST 2017, Petrozavodsk, Russia, August 28 - September 01, 2017, published at http://ceur-ws.org

1 LHC@Home global project

In 2002, as part of the ongoing search for ever better price-performance ratio computing, as CERN had moved from mainframes to workstations and then PCs, an article on the use of PlayStations suggested the use of even lower cost alternatives. Neither the PlayStation 2 nor 3, however, provided IEEE 754 compliant double precision floating-point arithmetic which was, and is, considered essential for most CERN applications. Instead an informal project Compact Physics ScreenSaver (CPSS) [1, 2] was established to attempt to use the several thousand Windows desktop PCs at CERN during nights and weekends when otherwise idle. It was then proposed to use the Berkeley Open Infrastructure for Network Computing (BOINC) infrastructure to extend the potential usage worldwide. Thus volunteer computing has been used successfully at CERN since 2004 with the LHC@Home project, and has provided additional computing power for CPU-intensive applications with small data sets, as well as an outreach channel for CERN activities. LHC@Home started off with the accelerator code SixTrack [3, 4] which had been successively ported from mainframe to supercomputer to emulator farms and PCs, and later on a gas detector simulation program, Garfield [5]. However, as applications running under BOINC had to be compiled for each and every possible client operating system, only the SixTrack application was ported to Windows, Linux and later MacOSX clients. Note that High Energy Physics (HEP) codes run almost exclusively under the Linux operating system.

Thanks to developments started at CERN, and later brought into the BOINC distribution, such Linux programs can now run on a Virtual Machine (VM) distributed to the volunteer computers via BOINC and running on volunteer PCs within the Oracle VirtualBox hypervisor. This use of virtualisation under BOINC was pioneered by the Test4Theory LHC@Home project during 2008-2011 [6, 7, 8, 9]. This development has allowed the LHC experiment collaborations to run their simulations also under BOINC, in the CernVM virtual machine. It is worth mentioning that use of Docker containers as a lighter alternative to virtual machines has been tested as a proof of concept, but this requires more work.

Several experimental groups have been running pilot BOINC projects for their collaborators to contribute simulations via BOINC and virtualisation. Following the experience with Test4Theory, ATLAS@Home and other pilot projects, with a view to include volunteer computing into the production computing infrastructure for HEP [10], a major effort has been undertaken to consolidate the original LHC@Home and host additional applications utilising virtualisation. Adding more applications to a BOINC project is straightforward; however to make multiple applications appeal to volunteers and users from different communities, application-specific credit was deployed. The accounts and BOINC credit of volunteers who had been contributing to the pilot projects Test4Theory/vLHCathome and ATLAS@Home were migrated to the consolidated LHC@Home project by means of a set of SQL scripts, as the information is all stored in the database. The volunteer's email address was used as unique key for the data, as the user id differs in each project depending on when the volunteer joined the BOINC project.

On the consolidated LHC@Home, users have a choice of applications that is enabled via LHC@Home project preferences. By default, only the SixTrack application (that does not require VirtualBox) is enabled for volunteers. Once registered, volunteers who wish to run other applications can enable e.g. ATLAS, CMS or Theory simulations via their LHC@Home project preferences. Today, active BOINC projects together harness about 7.5 Petaflops of computing power.

In terms of computing power provided by the volunteers, the average is about 1×10^5 simulation tasks. For SixTrack, peaks of 3.5×10^5 simultaneously running tasks on 2.4×10^4 hosts have been observed during SixTrack simulation campaigns, but note that every SixTrack task is run twice to eliminate random host errors and minimise the impact of a failing host. This can be compared against the average of 2.5×10^5 running batch jobs on 1.4×10^5 processor cores in the CERN computer centre, that is fully loaded with tasks of analysis and reconstruction of collisions recorded by LHC experiments, and has limited spare capacity for beam dynamics simulations. The applications of the LHC experiments that require virtualisation support on volunteer computers have operated with a sustained load of about 7000 tasks for ATLAS, 6000 for Theory, 3500 for LHCb, and 1000 for CMS.

1.1 SixTrack

SixTrack is an open source program for the simulation of charged particle trajectories in circular accelerators; it has been running under LHC@Home since 2004. Some 1.5×10^5 users with more than 3×10^5 PCs have been active LHC@Home volunteers since its launch. This has provided significant computing power for accelerator physics studies, for which there was no equivalent capacity available in the regular CERN computing clusters.

Volunteers contributing to SixTrack have delivered a sustained processing capacity of more than 45 TeraFlops. In Fig. 1 the time-evolution of the volunteers, active tasks, and cumulative number of working units (WU) since Feb. 2017. Note that each WU is submitted at least twice for ensuring numerical stability of the results. Note that the number of volunteers underestimates the actual CPU capacity available, as each volunteer could provide several machines and each machine might be multi-core.

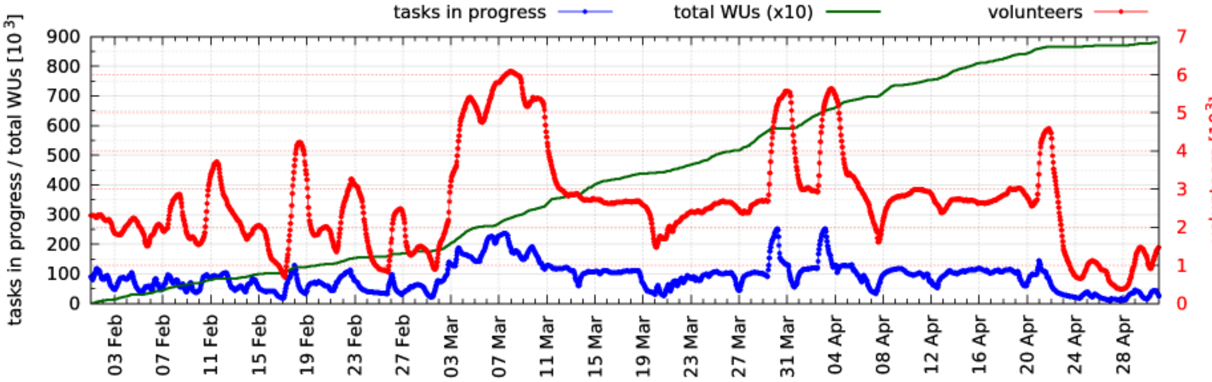

Figure 1: Time-evolution of the cumulative number of WUs, volunteers, and tasks sent to BOINC from Feb. 2017.

The SixTrack code is mainly Fortran-based, vectorized to take advantage of vector instructions, pipelining, and hardware features such as SSE and AVX. It was ported for use with BOINC to Windows, MacOSX and Linux by incorporating calls to the BOINC application programming interface (API) library and re-compiling and re-linking the source code to produce executables for each client platform. Since 2004, the application code has undergone several updates to adapt to new BOINC versions as well as to improvements to SixTrack itself (see [11] for a recent account on the code state). The principal functional changes for consistent and reliable operation are outlined in [12], but subsequent improvements now allow the use of several Fortran compilers, at any Fortran standard compliant level of optimisation, providing identical results, i.e. 0 Units difference in the Last Place (ULPs), on any IEEE 754 compliant hardware [13]. In order to achieve this, Fortran expressions, which could be evaluated in a different order as allowed by the standard, were parenthesised [14]. SixTrack can be built in many different configurations, e.g. for dynamic aperture (see Section 2) or collimation studies, and with or without support for checkpoint/restarting, compressed input/output, correct and consistent rounding of mathematical functions [15], BOINC, and more. Furthermore, it can run natively on most major platforms (Linux, MacOSX, Windows including XP, Free BSD, and Net BSD on x86 and x86_64 as well as Linux on AArch64 including Android systems), as long as a UNIX-like build environment is available; on Windows this is provided by MSYS2. The present CMake-base build system can compile from sources [16] and tests the reproducibility of the results using GNU, Intel, NAG Fortran compilers. Consistency down to 0 ULP is automatically verified between the versions, platforms, and compilers using a CTest-based test suite, which includes automatic building reports and test coverage published on CDash [17].

1.2 Theory

Since 2011, Monte-Carlo computer simulations of both ongoing and historical collider experiments have been performed in a CernVM virtual machine sent to volunteers using BOINC [6]. Such so-called “event-generator” programs (see [7] for an introduction and review) are used extensively in HEP, as explicit numerical models of the (often highly complicated) particle dynamics and to provide theoretical reference calculations for the experimental measurements. Via the Test4Theory project, which pioneered the use of virtual-machine technology for volunteer cloud applications, more than 3 trillion events have been simulated with different simulation programs. The generated events are compared to a large (and ever growing) library of particle-physics measurements, via the Rivet analysis preservation tool [18]. The results are stored as histograms and reference plots, in the online MCPlots database [9], which is available to the global particle-physics community. It is used by both the authors of the simulations and by their users, as a validation tool, and to guide further efforts to improve the physics models and to optimise their parameters (see e.g. [19]).

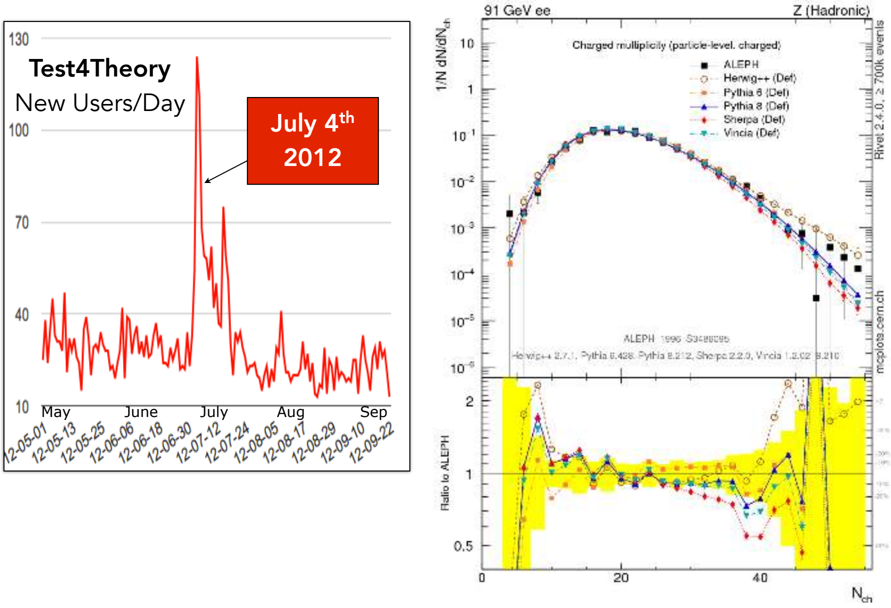

Figure 2: New users per day on Test4Theory during 2012 (left) and comparison of modern event generators to a legacy measurement (right, from the MCPlots web site [9]).

The left-hand pane of Fig. 2 shows a time slice from the summer of 2012, of the number of new users per day signing up for the Test4Theory project. On July 4th that year, CERN announced the discovery of the Higgs boson, prompting hundreds of new users to join the project. The right-hand pane shows one of the many thousands of plots that are available at the MCPlots site [9]. Several state-of-the-art models for particle collisions (coloured lines) are compared against an archived measurement performed in 1996 by the ALEPH experiment (black squares) [20], of the probability distribution for observing N charged particles (N_{ch} on the x axis) in electron-positron collisions at the LEP collider. (The lower pane shows the ratio of theory divided by data.) One clearly sees that the average of about 20 charged particles per collision is well reproduced by all the models, while their predictions differ in the tails of the distribution, where the uncertainty on the measurement (yellow band) was large.

1.3 A Toroidal LHC ApparatuS (ATLAS)

ATLAS@Home started in 2014 as an independent project where volunteers run Geant4 [21] Monte-Carlo simulation of particles passing through the ATLAS detector [22]. These simulations are well-suited to volunteer computing for several reasons: they involve less data transfer compared to other workloads; in ATLAS they are the largest consumer of CPU resources and hence there is always a reliable source of work; many simulation campaigns run over several months, so a fast turnaround is not expected.

ATLAS relies on virtualisation to allow its simulation software to run on non-Linux hosts. The CernVM project [23] provides virtual images tailored for the LHC experiments software and these images can run seamlessly inside the virtualisation layer provided by BOINC. ATLAS software is provided to the VM through the CernVM File System (CVMFS) [24], a remote read-only filesystem using aggressive local caching which is mounted inside the image. To avoid downloading the software every time the VM is started, the CVMFS cache inside the image is pre-filled with the required software, by running an example job, saving a snapshot of the image, and using that snapshot as the final image to distribute to volunteers.

One critical requirement when starting the project was that no sensitive ATLAS credentials should be distributed to volunteers. The solution was to use the model deployed in NorduGrid [25] and other environments such as High Performance Computing (HPC) centres which have restricted access to the outside world from the job worker nodes. The architecture of this model is shown in Fig. 3.

The Advanced Resource Connector (ARC) Computing Element (ARC CE) [26] takes care of data staging before and after the job runs and the ARC Control Tower (aCT) [27] provides the link with the ATLAS workload management system, PanDA [28]. Jobs which are assigned to ATLAS@Home by PanDA are picked up by the

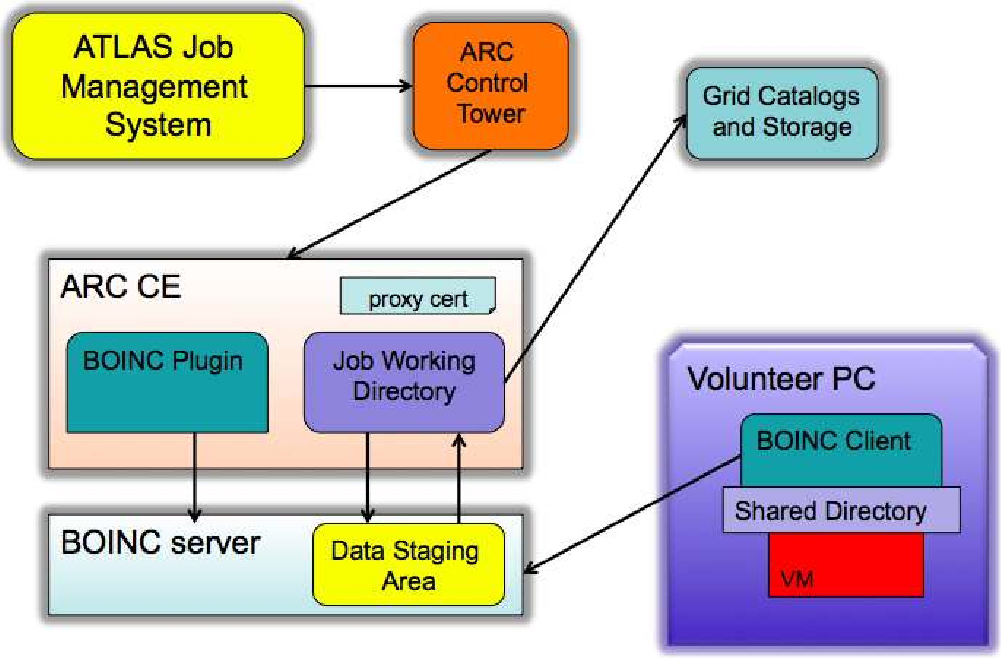

Figure 3: Architecture of ATLAS@Home.

aCT, and sent to an ARC CE connected to the BOINC server. ARC CE copies the required input files from Grid storage to a staging area inside the BOINC server. ARC CE supports many batch systems and a new plugin for a BOINC "batch system" was written to allow injection of jobs as work units in the BOINC server. Instead of calling batch system commands, this plugin uses the create_work command to inject jobs into the BOINC server and queries the BOINC database to find out when jobs have completed. The BOINC client on the volunteer's PC only has access to the BOINC server data staging area and no access to Grid storage or Grid credentials and so there is no chance of accidental or deliberate tampering with ATLAS data. Because ARC CE and aCT are services which are part of the regular ATLAS computing Grid, ATLAS@Home looks from the outside like a regular Grid site, which means no special treatment is needed when it comes to defining tasks, monitoring, accounting etc.

ATLAS@Home is one of the most demanding volunteer computing applications in part due to its high memory usage. A job using a single core can require a virtual machine with up to 2.5 GB of memory, and for many machines this means that it is not possible to fill all cores with ATLAS@Home tasks. However, ATLAS software can run on several cores inside a single virtual machine and can take advantage of sharing memory between processes running on each core. These multi-core jobs provide a significant memory saving, with an 8-core job typically using 5-6 GB of memory in total. Previously, BOINC only allowed a fixed memory limit per WU no matter how many cores were used. The ATLAS@Home jobs' memory requirements are dependent on the number of cores and so the project team implemented in BOINC a way of dynamically determining the memory required based on the number of cores. Two new parameters were added to the plan class, which describes the characteristics of the virtual machine. A base memory and memory per core can be specified and the memory of the virtual machine is calculated as base memory + (memory per core × number of cores). This feature was passed upstream and is now part of the standard BOINC software.

After the consolidation of Test4Theory, CMS and LHCb into a combined LHC project, it was obvious that ATLAS should follow. After a lengthy testing period the ATLAS app in LHC@Home was declared ready in March 2017, and after the final ATLAS@Home tasks finished a few weeks later, the users' credit was moved into LHC@Home. At the time of writing ATLAS volunteers have simulated almost 170 million ATLAS events (one event typically takes around 5 minutes of CPU time to simulate) and the combined resources add up to around 2% of overall ATLAS computing resources.

1.4 Compact Muon Solenoid (CMS)

CMS [29] is one of two general-purpose detectors at the LHC project, alongside ATLAS. Development began on a CMS@Home project in 2015 using a modified CMS Remote Analysis Builder v3 (CRAB3) [30] server VM submitting jobs running CMS standard software (CMSSW) [31] to a dedicated HTCondor [32] server VM rather than the normal submission to the Worldwide LHC Computing Grid (WLCG) [33]. The VMs were run at Rutherford Appleton Laboratory (RAL), UK.

Care was taken to match the type of jobs being run to the limitations of the volunteer environment. Of particular concern was the amount of data to be transferred since many users still have ADSL connections which may have upload speeds as low as 1 Mbps. This obviously ruled out analysis of CMS data, but still allowed the generation of Monte-Carlo simulations of collision events. Job parameters were adjusted to give average run-times of about one hour, and output files of the order of 50 MB. The BOINC server distributed tasks to run in the volunteers' VMs, and the tasks retrieved jobs from the HTCondor server, returning the results to a dedicated Data Bridge service [34] from where they could then be transferred to the normal CMS computing infrastructure. After a task has run for ∼12 hours, it terminates when the current job finishes.

As a comparison with standard Grid jobs, batches of 2×10^3 jobs consisting of 25 events producing top-antitop ($t\bar{t}$, or ttbar) pairs were submitted to both CMS@Home and the Grid. The number of result files received over time from subm

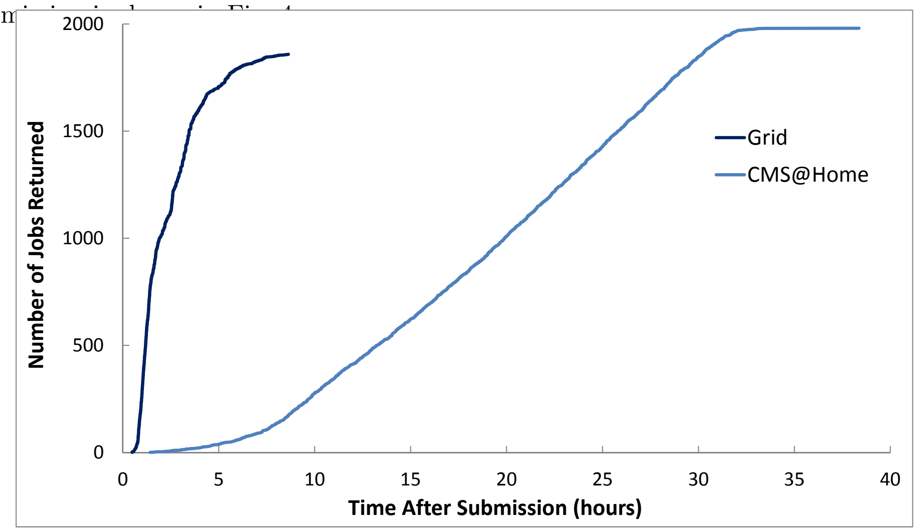

Figure 4: The distribution of result files received for 2×10^3 25-event ttbar simulation jobs, as a function of time from submission: dark curve – results from the Grid; light curve – results from CMS@Home volunteers.

Since the Grid has a large number of fast hosts, the first results started arriving after just 30 minutes, with 90% (1800) of the expected results received in about 6 hours. Unusually, 7.1% (142) of the result files were never received. Meanwhile, CMS@Home results began arriving after ∼80 minutes, but due to the small number of available volunteer hosts (∼100) only a limited number could run at any one time. Thus the graph of return times (Fig. 4) has an even slope for much of its duration as results returned at a constant rate. 90% of the results were received in 29.5 hours; in total 99% (1980) arrived in 38 hours.

As a test of a scientifically valuable process, the project turned to the simulation of the production of Λ_b^0 in LHC collisions, and its decay to a proton, a muon, and a neutrino. This is of interest as a background in measurements of a B_s decaying to two muons, since the proton may be misidentified as a muon. Because the Λ_b^0 is more massive (5.62 GeV/c^2) than B_s (5.37 GeV/c^2), the reconstructed mass of the $p + \mu$ overlaps the B_s mass spectrum, since the undetectable ν carries away a variable amount of energy. However, the production ratio is small, around 3×10^{-5}, so many proton-proton collisions need to be simulated to provide a significant number of desired events. Jobs simulating 2×10^5 collisions were used (median run-time 2h20m, result files ∼16 MB). In the last half of 2016, as the project developed and was incorporated into the larger LHC@Home, the number of simultaneous jobs increased and altogether several tens of billions of collisions were simulated, returning more than 2 million filtered events.

The project has now turned to the use of the work-flow management system (WMAgent) [35] for job submission. WMAgent gives the ability to specify a destination site within the CMS infrastructure to which results are automatically replicated using the transport software PhEDEx [36]. Thus fully end-to-end running of CMS Monte-Carlo production jobs has been demonstrated and the project will be able to contribute a significant computing resource to the CMS Collaboration. At the time of writing, volunteers are providing around 800 job slots to production, a figure that is expected to rise in the future.

1.5 Large Hadron Collider beauty experiment (LHCb)

The LHCb [37] experiment detector has been designed to filter out from the different particles generated by LHC those containing beauty and anti-beauty quarks (B-mesons) and the products of their decay. Unlike the other LHC experiments that surround the entire collision point with layers of sub-detectors, the LHCb detector extends along the beam pipe, with its sub-detectors piled behind each other. This is because the B-mesons do not travel in all directions, but rather stay close to the line of the beam pipe. Considering the growing need of computing power, the LHCb computing group has created a first prototype of the Beauty@Home project in 2013 to profit from volunteer computing resources.

The project uses the CERNVM Virtual Software Appliance [38], the BOINC framework, and the Distributed Infrastructure with Remote Agent Control (DIRAC) system for distributed computing [39, 40]. At the beginning the project was used only by users belonging to the LHCb Virtual Organisation. This because the architecture did not provide a secure technique to authenticate volunteers, but a trusted host certificate was contained in the machine dispatched to the volunteer.

The original problem was that pilot jobs needed to contact central DIRAC services such as the job matching or the job status update. They also needed to perform data management operations, such as the upload of the output files and the deployment of real credentials (proxy or server certificate), on untrusted machines, which was representing a big security hole. The necessity of having a secure authorization and authentication process to open the project to the outside world triggered the development of a DIRAC gateway service called Workload Management System Secure Gateway (WMSSecureGW). The service had the aim to interface untrusted volunteers machines to the DIRAC System authorizing BOINC users to execute LHCb jobs.

The WMSSecureGW service runs on a trusted machine, which has a valid certificate and accepts a dummy Grid certificate signed by a dummy certification authority (CA). The service receives all calls coming from the job and directed to different DIRAC services and it dispatches them as appropriate. Before the real storage upload is performed, the output data produced by the volunteer machines are uploaded on the gateway machine where a check has to be performed to avoid storing wrong data on LHCb storage resources. The architecture of the WMSSecureGW service is shown in Fig. 5.

Through this service the Beauty@Home has been integrated in the LHCb Grid infrastructure and the BOINC volunteers are running LHCb simulation jobs as all others Grid resources.

Currently, almost 3.5×10^3 simulation jobs are performed per day by volunteer computing resources, hoping that this number will grow in the near future, thanks to the increasing contribution of volunteers.

2 A SixTrack use case

2.1 CERN Large Hadron Collider and its High-Luminosity upgrade

Modern particle colliders are all based on superconducting magnets to generate high-magnetic field and hence high-energy beams. This class of magnets comes with intrinsic field errors that generate non-linear effects in the charged particle dynamics. Non-linearities are potentially harmful for particle's motion as they could drift away from the central trajectory, eventually hitting the beam pipe. This would induce beam losses or, even worse, a transition from the super- to normal-conducting state. Both events would entail an overall loss of accelerator performance. The only means to determine whether a charged particle will be eventually lost is via numerical simulations. The aim of these simulations is to determine the so-called dynamic aperture (DA), i.e. the region in phase space where the particle's motion is stable for a given number of turns.

Each simulation requires generating a set of initial conditions to be tracked through the accelerator structure for $10^5 - 10^6$ turns, which, in the case of the CERN Large Hadron Collider (LHC) corresponds to only 8-80 s out of a cycle of several hours. The DA depends on several physical parameters and scan over these quantities is essential to better understand the beam behaviour. Moreover, magnetic field errors are treated statistically and the DA computations are repeated for several realisations of these errors, typically 60, to ensure enough statistical relevance of the results. Overall, this implies that a typical study is made of $\approx 1 - 3 \times 10^6$ WUs each

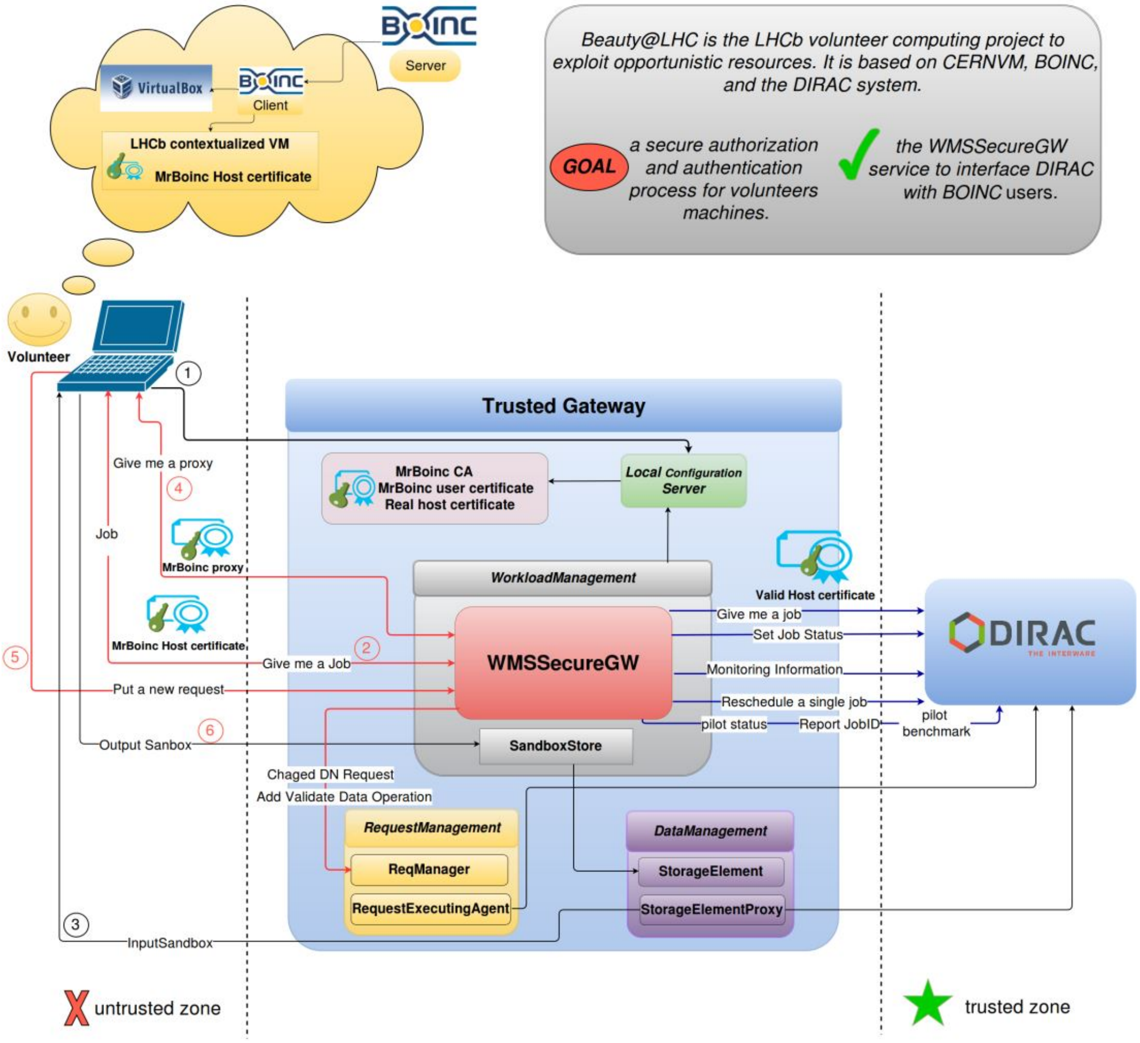

Figure 5: The whole gateway architecture, including the WMSSecureGW service and all services necessary to interface volunteers to the DIRAC framework.

performing tracking over $10^5 - 10^6$ turns. This makes LHC@Home the ideal system for DA simulations that, otherwise, would not be possible to perform on standard computing resources.

The limited number of turns that can be explored requires special techniques to extrapolate the particle behaviour to more relevant time scales [41] and dedicated measurement campaigns have been carried out to benchmark numerical simulations in the LHC without [42] and with [43] beam-beam effects. Examples of these studies are shown in Fig. 6, where the upper row shows comparison of measured and simulated DA, while in the lower row a typical scan of the extrapolated DA vs key parameters is shown.

For the LHC high-luminosity upgrade (HL-LHC) [44], beam simulations are essential for a reliable estimate of the collider's performance, also to guide the design of the new hardware. In Fig. 7 (left) the DA is shown as a function of the phase advance (horizontal and vertical) between the collision points in ATLAS and CMS, while (right) the DA as a function of transverse tunes including beam-beam interaction between bunches of 2.2×10^{11} protons is depicted (see also [45]). Note that these studies are essential to select the parameters' values providing the maximum DA, hence optimising the accelerator's design.

2.2 Future challenges

The CERN Future Circular Collider (FCC) [46], a 100 TeV centre-of-mass energy collider, is one of the options for future large-scale particle physics experiments [47]. Design studies involving world-wide collaborative efforts

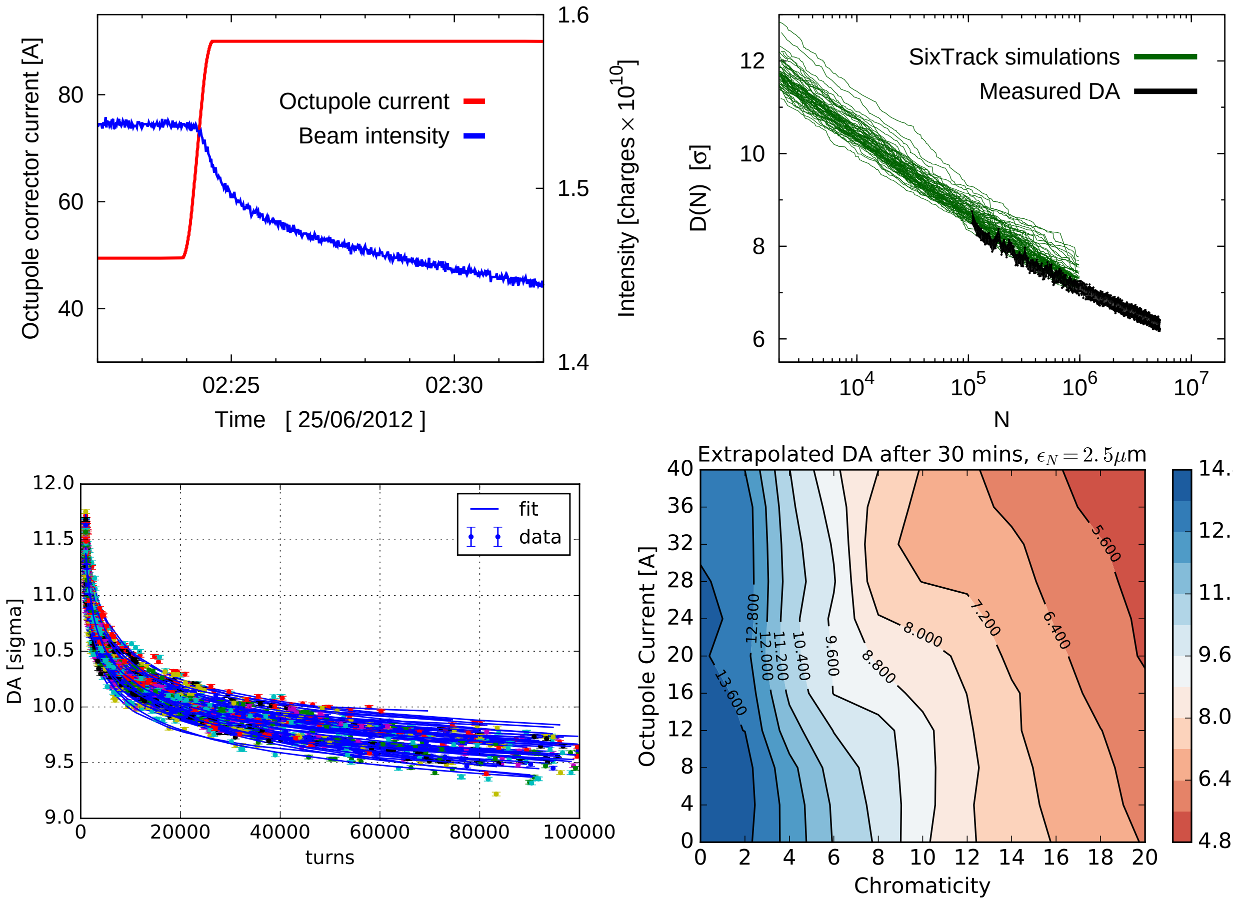

Figure 6: Upper left: measured beam intensity evolution during an experimental session. Upper right: comparison between simulated and measured DA of the LHC at injection. Lower left: DA evolution with the number of turns from SixTrack simulations compared to fits of the data for the individual seeds. Lower right: Extrapolated DA of LHC at 30 minutes after injection as a function of different chromaticities and octupole settings.

are in full swing. FCC is a true challenge, both in terms of accelerator physics as well as from the computational standpoint and the huge capacity offered by volunteer computing is an added value.

The study of the evolution of distributions of initial conditions to mimic a real beam is a challenge ahead of us. This could address questions concerning collective instabilities in presence of beam-beam effects [48, 49] or the losses induced by the interaction between the beam and the jaws of collimators used to clean the beam halo [50]. The computing capacity is beyond the capabilities of standard facilities and volunteer computing would be the ideal solution.

3 Conclusions and Outlook

Volunteer computing with BOINC has proven to bring in significant resources for simulations for accelerator physics and HEP community. Thus expanding the number of volunteers taking part to LHC@Home is our long-term goal.

The computational problem in accelerator physics is largely one of throughput and the number of processors available is more important than the per processor performance. Therefore, by providing support for ARM processors with Android (tablets and smart phones) and for Raspberry Pi, an even larger number of processors can be made available for at least the SixTrack application. We are also working on the porting of the SixTrack application to use GPU resources. In fact, since most computers used by volunteers have graphics processors, usage of GPUs might generate an estimated five- to ten-fold increase of the throughput of SixTrack jobs running on same number of volunteers' computers.

SixTrack is undergoing major development efforts to open up new domains of accelerator physics, needed for

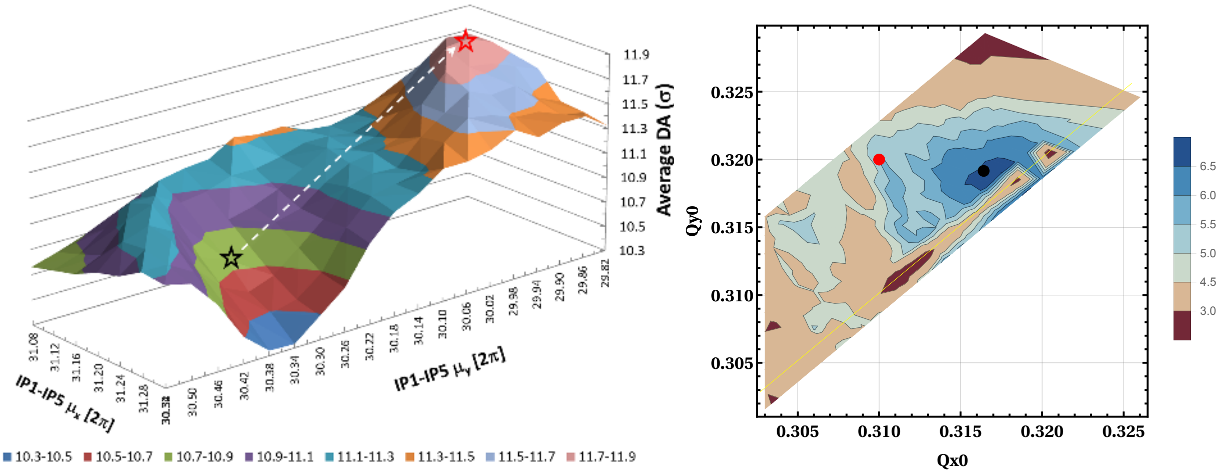

Figure 7: Left: DA averaged over the 60 realisation of the magnetic field errors as a function of the phase advance (horizontal and vertical) between the collision points in ATLAS and CMS. Right: DA as a function of transverse tunes including beam-beam interaction between bunches of 2.2×10^{11} protons.

a better understanding of current and future circular particle colliders. LHC@Home is the ideal environment to exploit at best the new code capabilities in view of massive numerical simulations.

Thanks to virtualisation, the scope of applications that may run under BOINC has been widened. As use of virtualisation with volunteer computing implies more complexity and overhead for the volunteers, potential simplification, e.g., with container technology, should be investigated further.

The volunteer computing strategy at CERN is to integrate the volunteer computing tool chain with the HTCondor batch system used for computing on batch, cloud, and Grid computing resources. This approach will make it easier for scientists to submit work to different resources allowing the IT team to direct workloads to the appropriate ones. In this respect, further attention is needed to evolve the BOINC middleware and to improve the integration with HTCondor. An effort to evolve the BOINC community software with contributions from major BOINC projects and stakeholders is required to ensure a long-term future for BOINC and the current volunteer computing community.

Acknowledgements

Our warm thanks go to all people that supported and continue to support us by donating CPU capacity, which is a huge contribution to our studies! We hope that even more volunteers will join LHC@Home to help us pushing further the detail of the studies that we can perform.

We gratefully acknowledge the contributions made to the CMS@Home development by the CMS Collaboration in providing the many software packages used, especially CMSSW, CRAB3, WMAgent and PhEDEx. We also acknowledge financial support from the Science and Technology Facilities Council, UK.

We also gratefully acknowledge the support by the European Circular Energy-Frontier Collider Study, H2020 under grant agreement no. 654305 and by the Swiss State Secretariat for Education, Research and Innovation SERI.

References

[1] E. McIntosh and A. Wagner, CERN Modular Physics Screensaver or Using Spare CPU Cycles of CERN's Desktop PCs, in Proc. 14th International Conference on Computing in High-Energy and Nuclear Physics (CHEP 2004), p. 1055 (2004).

[2] `http://cpss.web.cern.ch/cpss/`

[3] F. Schmidt, SixTrack Version 4.5.17, Single Particle Tracking Code Treating Transverse Motion with Synchrotron Oscillations in a Symplectic Manner, User's Reference Manual, CERN/SL/94-56 (AP).

[4] `http://sixtrack.web.cern.ch/sixtrack-ng/`

[5] `http://garfield.web.cern.ch/garfield/`

[6] D. Lombrana Gonzalez *et al.*, Virtual machines & volunteer computing: Experience from LHC@Home: Test4Theory project, PoS ISGC2012 036 (2012).

[7] A. Buckley *et al.*, General-purpose event generators for LHC physics, Phys. Rept. **504**, 145 (2011), `http://www.montecarlonet.org`

[8] N. Høimyr *et al.*, BOINC service for volunteer cloud computing, J. Phys.: Conf. Ser. **396** 032057 (2012).

[9] A. Karneyeu, L. Mijovic, S. Prestel and P. Z. Skands, MCPLOTS: a particle physics resource based on volunteer computing, Eur. Phys. J. C **74**, 2714 (2014), `http://mcplots.cern.ch`

[10] N. Høimyr *et al.*, Towards a Production Volunteer Computing Infrastructure for HEP, J. Phys.: Conf. Series **664** 022023 (2015).

[11] K. Sjobak, R. De Maria, E. McIntosh, A. Mereghetti, J. Barranco, M. Fitterer, V. Gupta and J. Molson, "New features of the 2017 SixTrack release", presented at the 8th Int. Particle Accelerator Conf. (IPAC'17), Copenhagen, Denmark, May 2017, paper THPAB047.

[12] E. McIntosh, F. Schmidt and F. de Dinechin, Massive Tracking on Heterogeneous Platforms, in Proc. of 9th International Computational Accelerator Physics Conference, DOI: 10.5170/CERN-2005-002, p. 13 (2006).

[13] E. McIntosh, paper in preparation.

[14] H. Renshall, private communication.

[15] C. Daramy, D. Defour, F. de Dinechin, J.-M. Muller, CR-LIBM: a correctly rounded elementary function library, Proc. SPIE 5205, Advanced Signal Processing Algorithms, Architectures, and Implementations XIII, 458 (December 31, 2003); doi:10.1117/12.505591

[16] SixTrack source repository http://github.com/SixTrack/SixTrack

[17] https://abp-cdash.web.cern.ch/abp-cdash/index.php?project=SixTrack

[18] A. Buckley *et al.*, Rivet user manual, Comput. Phys. Commun. **184**, 2803 (2013).

[19] P. Skands, S. Carrazza and J. Rojo, Tuning PYTHIA 8.1: the Monash 2013 Tune, Eur. Phys. J. C **74**, no. 8, 3024 (2014).

[20] R. Barate *et al.* [ALEPH Collaboration], Studies of quantum chromodynamics with the ALEPH detector, Phys. Rept. **294**, 1 (1998).

[21] S. Agostinelli *et al.*, Geant4 - A Simulation Toolkit, Nucl. Instrum. & Methods A **506** 250-303 (2003).

[22] ATLAS Collaboration, The ATLAS Experiment at the CERN Large Hadron Collider, J. Inst. 3 S08003 (2008).

[23] P. Buncic *et al.*, CernVM - a virtual software appliance for LHC applications, J. Phys.: Conf. Series **219** 042003 (2010).

[24] C. Aguado Sanchez *et al.*, CVMFS - a file system for the CernVM virtual appliance, Proc. XII Advanced Computing and Analysis Techniques in Physics Research (PoS(ACAT08)012) p. 52 (2008).

[25] NorduGrid Collaboration, `http://www.nordugrid.org`

[26] M. Ellert *et al.*, Advanced Resource Connector middleware for lightweight computational Grids, Future Gener. Comput. Syst. **23** pp. 219240 (2007).

[27] A. Filipcic on behalf of ATLAS Collaboration, arcControlTower: the System for Atlas Production and Analysis on ARC, J. Phys.: Conf. Series **331** 072013 (2011).

[28] T. Maeno on behalf of ATLAS Collaboration, PanDA: distributed production and distributed analysis system for ATLAS, J. Phys.: Conf. Series **119** 062036 (2008).

[29] CMS collaboration, The CMS experiment at the CERN LHC, J. Inst. **3**, S08004 (2008).

[30] M. Mascheroni *et al.*, CMS distributed data analysis with CRAB3, J. Phys.: Conf. Series **664**, 062038 (2015).

[31] C.D. Jones, M. Paterno, J. Kowalkowski, L. Sexton-Kennedy and W. Tanenbaum, The New CMS Event Data Model and Framework, in Proc. 15th International Conference on Computing in High-Energy and Nuclear Physics (CHEP 2006).

[32] B. Bockelman, T. Cartwright, J. Frey, E.M. Fajardo, B. Lin, M. Selmeci, T. Tannenbaum and M. Zvada, Commissioning the HTCondor-CE for the Open Science Grid, J. Phys.: Conf. Series **664**, 062003 (2015).

[33] `http://wlcg-public.web.cern.ch/`

[34] `https://cds.cern.ch/record/2002390`

[35] E. Fajardo, O. Gutsche, S. Foulkes, J. Linacre, V. Spinoso, A. Lahiff, G. Gomez-Ceballos, M. Klute and A. Mohapatra, A new era for central processing and production in CMS, J. Phys.: Conf. Series **396**, 042018 (2012).

[36] A. Sanchez-Hernandez, R. Egeland, C-H. Huang, N. Ratnikova, N. Maginie and T. Wildish, From toolkit to framework - the past and future evolution of PhEDEx, J. Phys.: Conf. Series **396**, 032118 (2012).

[37] LHCb Collaboration, The LHCb Detector at the LHC, J. Inst. **3**, S08005 (2008).

[38] P. Buncic *et al.*, CernVM - a virtual appliance for LHC applications, in Proceedings of the XII. International Workshop on Advanced Computing and Analysis Techniques in Physics Research, Erice, 2008 PoS(ACAT08)012.

[39] A. Tsaregorodtsev *et al.*, DIRAC: a community Grid solution, J. Phys.: Conf. Series **119** 062048 (2008).

[40] `http://diracgrid.org`

[41] M. Giovannozzi, Proposed scaling law for intensity evolution in hadron storage rings based on dynamic aperture variation with time, Phys. Rev. ST Accel. Beams **15**, 024001 (2012).

[42] E. Maclean, M. Giovannozzi, R. Appleby, Novel method to measure the extension of stable phase space region of proton synchrotrons using nekoroshev-like scaling laws, in preparation.

[43] M. Crouch *et al.*, Dynamic aperture studies of long-range beam-beam interactions at the LHC, IPAC17, Copenhagen, Denmark, paper THPAB056.

[44] G. Apollinari *et al.*, High-Luminosity Large Hadron Collider (HL-LHC): Preliminary Design Report, CERN, Geneva, 2015, `https://cds.cern.ch/record/2116337/`

[45] T. Pieloni, D. Banfi, J. Barranco, Dynamic Aperture Studies for HL-LHC with beam-beam effects, CERN-ACC-NOTE-2017-0035 (2017).

[46] FCC design studies, `https://fcc.web.cern.ch/`

[47] M.Kramer, The update of the European strategy for particle physics, Phys. Scr. 2013 014019 (2013).

[48] X. Buffat *et al.*, Stability Diagrams of colliding beams, Phys. Rev. ST Accel. Beams **17** 111002 (2014).

[49] C. Tambasco *et al.*, Impact of incoherent effects on the Landau Stability Diagram at the LHC, IPAC17, Copenhagen, Denmark, paper TUPVA031.

[50] A. Mereghetti *et al.*, SixTrack for Cleaning Studies: 2017 Updates, IPAC17, Copenhagen, Denmark, paper THPAB046.