Efficient precision simulation of processes with many-jet final states at the LHC

Enrico Bothmann,[1] Taylor Childers,[2] Christian Gütschow,[3] Stefan Höche,[4] Paul Hovland,[2] Joshua Isaacson,[4] Max Knobbe,[1] and Robert Latham[2]

[1]*Institut für Theoretische Physik, Georg-August-Universität Göttingen, 37073 Göttingen, Germany*
[2]*Argonne National Laboratory, Lemont, Illinois 60439, USA*
[3]*Centre for Advanced Research Computing, University College London, Gower Street, London, WC1E 6BT, United Kingdom*
[4]*Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA*

(Received 29 September 2023; accepted 28 November 2023; published 17 January 2024)

We present a scalable technique for the simulation of collider events with multijet final states, based on an improved parton-level event file format and scalable I/O. The method is implemented for both leading- and next-to-leading-order QCD calculations. We perform a comprehensive analysis of the computing performance and validate our new framework using Higgs-boson plus multijet production with up to seven jets. We make the resulting code base available for public use.

DOI: 10.1103/PhysRevD.109.014013

I. INTRODUCTION

The simulation of events with high-multiplicity final states in experiments at the Large Hadron Collider (LHC) is a challenging computational problem [1–3]. Using the best available algorithms, the calculation of the integrand for multijet processes scales at best exponentially with increasing particle multiplicity. The integration over the many-body phase space calls for suitable importance-sampling techniques, which often also scale exponentially [4–14]. Hence, while being a solved problem, in principle, the calculation of cross sections and the production of unweighted events at high jet multiplicity is still a hard problem to date. Computing techniques have remained conceptually identical since their inception four decades ago and typically make use of dynamic programming [14–23].

While the calculation of hard cross sections with full quantum interference effects is a considerable challenge even at tree level, the Markovian methods used in parton showers are often sufficient to describe the dynamics of collider events, and in fact, they are necessary to properly account for the all-orders resummation of virtual corrections [1,24,25]. The combination of the evolution implemented in parton showers with the exact calculations implemented by hard matrix elements provides the best available physics modeling of LHC events, accounting for both interjet correlations and intrajet evolution by means of next-to-leading order (NLO) matching [26–30] and multijet merging [31–44]. However, parton showers and the subsequent hadronization [45–50] and multiple interaction [51–53] models are associated with various free parameters. Varying these parameters is key to assessing the uncertainty of LHC simulations.

The large difference in computation time required for parton- and particle-level simulations [54,55], and the need to perform simulations for multiple parton shower, underlying event, and hadronization parameters in order to assess theory systematics, makes it natural to separate the generation of LHC events into the calculation of the hard interaction and the simulation of the remaining physics aspects. A number of approaches have been proposed to address this problem. The earliest and most widely used include the user process functionality of PYTHIA [56], and the Les Houches event file strategy [57–59]. More recently, with the need for high-statistics event simulation leading to the use of high-performance computing facilities [60], the need for improved I/O performance and scalable I/O has become apparent. In this manuscript, we report on the extension of a new Monte-Carlo (MC) event generation framework aimed at solving this problem [54]. First, we enable the handling of the standard and hard events needed for next-to-leading matching in the MC@NLO method. Second, we propose a new layout of the event file format at the core of the framework, in order to increase the performance in large-scale parallel processing. Third, we implement the new technology in various parton- and particle-level event generators. At parton level we use SHERPA [61,62] with the two internal matrix-element generators Amegic [63] and Comix [14], as well as the new

platform agnostic leading-order parton-level event generator Pepper [13,64,65]. At particle level, we use the event generators PYTHIA [56,66] and SHERPA [61,62]. In addition to solving the problem of scalability, the possibility to combine different particle-level simulations with the exact same perturbative input offers unprecedented opportunities for the study of theory systematics.

We also discuss a first phenomenological application of our new algorithms. We simulate Higgs-boson plus multijet events with up to seven jets at tree level and up to two jets at next-to-leading-order QCD precision. With the High-Luminosity LHC expected to collect 3 ab^{-1} of data, these predictions can be used to test QCD associated Higgs production over a large dynamic range. Moreover, Higgs-boson plus multijet events play an important role as irreducible backgrounds to more detailed tests of the Higgs sector of the Standard Model, especially in weak vector boson fusion. Using the MEPS@NLO merging method [40,41], in particular, the reweighting of higher-multiplicity tree-level predictions with the help of Born-local K factors from the Higgs plus two-jet setup, our new code base enables previously the most precise predictions of Higgs plus ≥ 4 jet events at low Higgs-boson transverse momentum. We make the corresponding input event samples publicly available [67,68] and provide the parton- and particle-level event generators that can be used to generate and process these event files.

This manuscript is structured as follows: Sec. II discusses the challenges faced in previous simulation campaigns and provides a documentation and performance assessment of our new event file format and parallel event generation techniques, including the changes needed for the simulation of events at next-to-leading-order QCD precision. Section III discusses the parton-level components of the framework, and Sec. IV focuses on the particle-level components. Section V presents the first phenomenological application and discusses the impact of NLO matching and multijet merging in Higgs plus multijet production. We conclude with an outlook in Sec. VI.

II. EXTENDED PARALLELIZATION FRAMEWORK

The first scalable particle-level event generator was based on legacy versions of Alpgen [21] and PYTHIA 6 [56] and was introduced in Ref. [60]. To make state-of-the art parton- and particle-level simulation tools available for use on high-performance computing (HPC) systems, Ref. [54] proposed a new event generation framework, based on SHERPA2 [62] and PYTHIA 8 [66]. This framework is based on a parallelized main routine for PYTHIA 8 and a new parton-level event file format, using the HDF5 library for parallel I/O [69,70]. It solved the main problem of making the event production scalable to thousands of message passing interface (MPI) ranks, but still suffers from an I/O bottleneck. Various modern high-performance computing systems are not well suited for the fast writing and reading of large amounts of data, as is common in parton-level event simulations. The solution to this problem is discussed in this section.

Reference [54] also did not provide a means to store information for parton-level events simulated at NLO QCD precision. At present, a pure leading-order-based event simulation falls short of the precision requirements at the LHC experiments. Therefore, an extension of the previous simulation framework to NLO QCD precision is an additional problem we will address. At leading-order QCD, one can write the expectation value of an arbitrary infrared safe observable O at particle level as

$$\langle O \rangle = \int \mathrm{d}\Phi_B \mathrm{B}(\Phi_B) \mathcal{F}_{\mathrm{MC}}(O, \Phi_B), \tag{1}$$

where B is the differential Born cross section, including flux and symmetry factors, as well as the parton luminosity, and $\mathrm{d}\Phi_B$ is the differential phase-space element, including the integration over the light-cone momentum fractions of the initial-state partons. The functional $\mathcal{F}_{\mathrm{MC}}(O, \Phi_B)$ implements the parton shower and is explained in more detail in [26,29]. In the MC@NLO [26] or POWHEG [27] NLO QCD matching technique, Eq. (1) becomes

$$\begin{aligned}\langle O \rangle = \int \mathrm{d}\Phi_B \Bigg[& \mathrm{B}(\Phi_B) + \mathrm{V}(\Phi_B) + \mathrm{I}(\Phi_B) + \int \mathrm{d}z_1 \mathrm{d}z_2 \mathrm{KP}(\Phi_B, z_1, z_2) \\ & + \sum_{ijk} \int \mathrm{d}\Phi_{+1,ijk} (\mathrm{D}^{(A)}_{ijk}(\Phi_B, \Phi_{+1,ijk}) - \mathrm{D}^{(S)}_{ijk}(\Phi_B, \Phi_{+1,ijk})) \Bigg] \mathcal{F}_{\mathrm{MC}}(O, \Phi_B) \\ & + \int \mathrm{d}\Phi_R \left(\mathrm{R}(\Phi_R) - \sum_{ijk} \mathrm{S}(\Phi_{B,ijk}(\Phi_R)) \right) \mathcal{F}_{\mathrm{MC}}(O, \Phi_R),\end{aligned} \tag{2}$$

where V and R are the virtual and real-emission corrections, S and I are the differential and integrated NLO infrared subtraction counterterms, and KP are the factorization scale-dependent finite corrections arising from the combination of the integrated infrared subtraction counterterms with collinear mass-factorization counterterms. The infrared subtraction counterterms are most commonly defined in the Frixione-Kunszt-Signer [71] or the Catani-Seymour [72,73] subtraction scheme and depend on the momenta and flavors of two partons i and j that are to be combined, as well as a

spectator or recoil momentum k. The differential phase-space element for the real-emission process is given by $\mathrm{d}\Phi_R$. It can be factorized into a differential Born phase-space element and a single-emission phase-space element as $\mathrm{d}\Phi_R = \mathrm{d}\Phi_{B,ijk}\mathrm{d}\Phi_{+1,ijk}$ [29].

The NLO QCD expression (2) shows a number of important differences compared to the leading-order (LO) expression (1).

(i) There are two different classes of events, one with parton-shower starting condition Φ_B (the so-called standard events, or $\mathbb{S}$ events) and one with parton-shower starting condition Φ_R (the so-called hard events, or $\mathbb{H}$ events).
(ii) The $\mathbb{H}$ events require a simple, leading-order-like phase-space generator for $\mathrm{d}\Phi_R$, which implies that the event file format is the same as at leading order.
(iii) The $\mathbb{S}$ events require two additional integrations, one for the KP counterterms and one for the one-emission phase space $\mathrm{d}\Phi_{+1,ijk}$. They also require a sampling of the indices i, j, and k.

In order to make MC@NLO $\mathbb{S}$ events reproducible and enable a reweighting of events to arbitrary parton distribution function (PDF) sets and/or scales at NLO QCD, the event format therefore requires the storage of the indices i, j, and k, as well as the phase-space point $(\Phi_B, \Phi_{+1,ijk})$ and the momentum fractions z_1 and z_2. Note that this information is sufficient to encode the quantities needed for exact recovery of the Monte Carlo weights not only in the SHERPA framework, but also in the POWHEG and aMC@NLO generators. We will define the corresponding datasets in Sec. II A.

A. Event file format

In this subsection we describe the new event file layout, which includes the optimizations that will be described in Sec. II B as well as extensions for event simulation at NLO QCD. Both are inspired by Les Houches event files (LHEF) standard [57], which is widely used in the high-energy physics community. The LHEF format is based on XML, which makes it flexible enough to add any desired feature, but poses a challenge for I/O operations at scale. The HDF5 format instead uses a computing model similar to databases, making it rigid, but highly efficient in parallel workflows. In general, the performance of file input and output operations depends on both throughput and latency. Each additional dataset that is added to an HDF5 file will incur a performance penalty in read and write operations, because of the added latency associated with dataset access. This latency can be much more relevant for I/O performance at scale than the throughput. In practice, it is therefore desirable to use as few datasets as possible and to consolidate the properties of events and particles, even if that means using a data type that naively does not match the property which it is supposed to store (such as a float to store the particle ID). As a consequence, we choose double precision numbers to represent nearly all properties stored in the file.

The LHEF format comprises global properties as well as eventwise properties [57]. Global properties include process information (i.e., the type of collisions) and total cross sections, as well as reweighting information. The eventwise properties are the process ID, the event weight, and the scale of the hard process, as well as the values of α_{QCD}, α_{QED}, and the list of particles generated. The latter contain information about momentum four-vectors, particle ID, charge, spin, and lifetime, as well as production history. We collect this information in consolidated datasets, reflecting the global, eventwise, and particle properties. In addition, we introduce two new datasets, which include the event- and particlewise information needed to simulate NLO QCD events in the MC@NLO matching scheme. We will call this structure the LHEH5 event file format [74].

We follow the naming scheme of Ref. [54] and define datasets called INIT and PROCINFO that are used to store basic information about the entirety of events contained in the file. We also add a new dataset, VERSION that identifies the version of the event file format. Eventwise properties for leading-order events are stored in the dataset EVENTS and for MC@NLO $\mathbb{S}$ events in the dataset CTEVENTS. Particlewise properties for leading-order events are stored in the dataset PARTICLES and for MC@NLO $\mathbb{S}$ events in the dataset CTPARTICLES. Each dataset is a two-dimensional array and has an HDF5 attribute PROPERTIES that identifies the individual columns of the dataset in order of appearance. For example, the PROCINFO dataset has the properties procID, npLO, npNLO, xSection, error, and unitWeight. In future updates of the event file format, these attributes can be used to communicate the content of the individual entries to the user of the file, similar, although not quite as flexible, as in the case of XML-based Les Houches event files. The content of all datasets is summarized in Table I. In addition to Ref. [54], we introduce the following entries:

(1) Process properties npLO and npNLO. If the process is computed at leading-order QCD, we set npLO to the final-state particle multiplicity. If the process is computed at next-to-leading-order QCD, we instead set npNLO to the final-state particle multiplicity at Born level.
(2) At NLO QCD, we include the minimal information needed to reconstruct the complete event weight in the MC@NLO matching method. For hard remainder events ($\mathbb{H}$ events), the leading-order type information is sufficient. For standard events ($\mathbb{S}$ events) we add the following:
   (a) Counterterm properties in the CTEVENTS dataset: IJT and KT refer to the Born-level QCD dipole used to generate a real-emission phase-space point in Eq. (2), I, J, K correspond to the respective particle IDs at real-emission level.

TABLE I. Datasets in the LHEH5 event format.

Name	Data type	Contents
VERSION	3 × int	Version ID
INIT	10 × double	beamA, beamB, energyA, energyB, PDFgroupA, PDFgroupB, PDFsetA, PDFsetB, weightingStrategy, numProcesses
PROCINFO	6 × double	procId, npLO, npNLO, xSection, error, unitWeight
EVENTS	9 × double	pid, nparticles, start, trials, scale, fscale, rscale, aqed, aqcd
PARTICLES	13 × double	id, status, mother1, mother2, color1, color2, px, py, pz, e, m, lifetime, spin
CTEVENTS	9 × double	ijt, kt, i, j, k, z1, z2, bbpsw, tlpsw
CTPARTICLES	4 × double	px, py, pz, e

TLPSW is the phase-space weight $d\Phi_B$ and BBPSW is the corresponding single-emission phase-space weight $d\Phi_{+1,ijk}$. The variables z1 and z2 are the MC points of the integration variables in the KP contribution. See Eq. (2) for details.

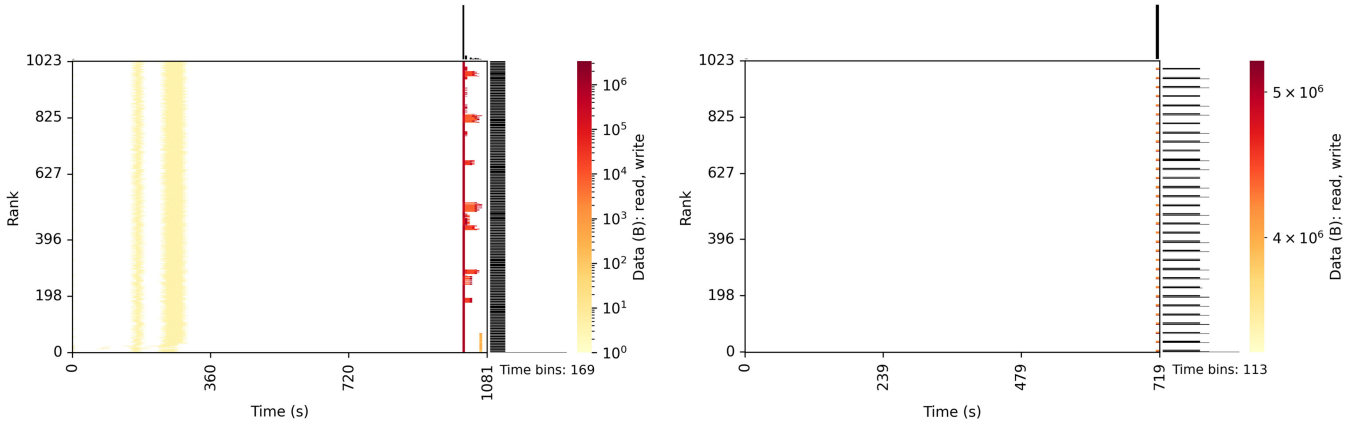
(b) Counterterms properties in the CTPARTICLES dataset: PX, PY, PZ, and e store the momenta of all particles in the phase-space point $(\Phi_B, \Phi_{+1,ijk})$. See Eq. (2) for details.

B. I/O operations at scale

Our event simulation frameworks read and write event data through a multilayered I/O software stack based on HDF5 [75], an array-oriented library and data model, which in turn uses MPI-IO [76]. Within SHERPA and PYTHIA, HDF5 is accessed through the HighFive header library [77]. Each layer typically provides tuning parameters. Optimal performance can, in principle, be achieved with the help of sophisticated I/O tuning systems [78]. However, selecting the right parameters with the help of subject expertise is often more efficient and more reliable. In our case, some straightforward changes to the I/O layer resulted in large performance gains, reducing the I/O for particle-level events to a very small fraction of the overall run-time.

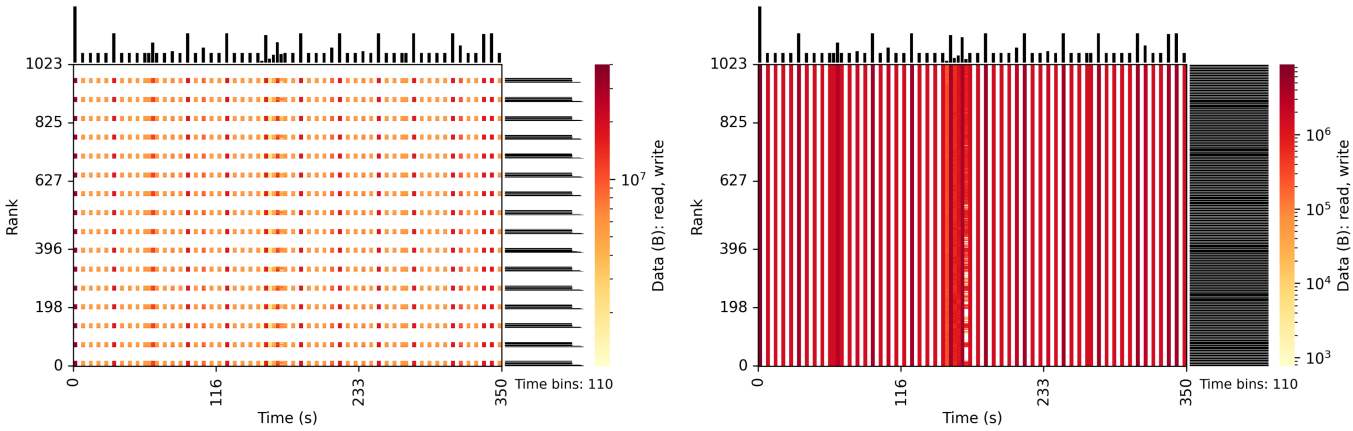
Figure 1 shows profiling results obtained with the help of Darshan [79,80] for a parton-level simulation of Higgs plus four jets at leading order, run on 1024 ranks of the Cori system at NERSC.[1] Here we make use of a new feature of Darshan, depicting which ranks perform I/O at which times. The color in the heat map represents the transferred data volume. Time runs along the x axis, MPI ranks along the y axis. A histogram along the top axis reports the total data volume over time. A histogram along the right axis reports the total data volume per rank. Figure 1(a) depicts the initial performance of our simulation framework for parton-level event production. As described in [54], the near lockstep event production implies that I/O also occurs in locksteps, making it a good candidate for collective I/O operations. These are operations where all ranks of a parallel computing job are writing to/reading from an output file at the same time and where it is guaranteed that the I/O operations are already synchronized among the jobs, such that no further coordination is needed by the I/O library. An example of this kind would be a number of MPI ranks, each writing to/reading from a different slice of a file. Figure 1(a) shows that the major output operations start at the same time (about 1000 s after startup) on the various ranks, but they end at very different times, being responsible for a large variation in run-time overall. This is reflected in the different length of the red bars near the right of the heat map (output occurring near the end of the run). In addition, all ranks perform actual POSIX operations, putting unnecessary strain on the Lustre file system. Finally, there are uncoordinated I/O operations during the run (at about 160 and about 260 s after startup), which are visible as the light yellow areas in the heat map. Again, all MPI ranks participate in POSIX operations, putting unnecessary stress on the Lustre file system. The total time spent in I/O operations was 33.5% in this case and was mostly due to file access coordination. Figure 1(b) shows the I/O after optimizations. Collective operations are enabled, for which we make use of an updated HighFive library, exposing HDF5's collective data and collective metadata features. The histogram running along the right of the plot shows that there are now very few ranks participating in I/O at the POSIX level, and that the start and end times are nearly identical on all ranks. The overall I/O activity occurs only at the end of the run and is barely noticeable. This level of performance was achieved only after consolidating the individual HDF5 datasets proposed in [54] into a single dataset, which greatly reduced latency, cf. the comments at the beginning of Sec. II A. We also limited stat calls performed by the program to the master rank and broadcast the results of the call via MPI. The total

[1]Cori was a Cray XC40 system, composed of 2388 Intel Xeon "Haswell" processor nodes, 9688 Intel Xeon Phi "Knight's Landing" nodes, and a Cray Aries network with Dragonfly topology with > 45 TB/s global peak bisection bandwidth. https://nersc.gov/systems/cori.



(a) POSIX operations before optimization (b) POSIX operations after optimization

FIG. 1. Darshan graphs depicting the POSIX I/O behavior of SHERPA during parton-level event generation for $H + 3$ jets at leading-order QCD on 1024 MPI ranks. (a) Before optimizations. I/O operations are fragmented and uncoordinated among processes. The total I/O cost is 33.5% of the run-time. (b) Including collective I/O, improved file layout to reduce metadata operations and limiting stat calls to the master rank. The total I/O cost is below 1 s. The number of events was held constant for this test, and the total amount of data written was 1.01 GiB.



(a) POSIX operations (b) MPI-IO operations

FIG. 2. Darshan graphs depicting the I/O behavior of SHERPA during particle-level event generation for $H + 4$ jet leading-order multijet merging on 1024 MPI ranks. (a) POSIX I/O behavior. (b) MPI-IO behavior. The time spent in I/O operations was less than 5% of the run-time. All figures after including collective I/O improved file layout to reduce metadata operations and limit stat calls to the master rank. The number of events was held constant for this test, and the total amount of data read was 128.85 GiB.

I/O time was thus reduced to a negligible amount, below 1 s per rank. This concluded our optimization of the parton-level simulation.

Figure 2 shows profiling results obtained with the help of Darshan [79,80] for a particle-level simulation of Higgs plus four jets with leading-order multijet merging, run on 1024 ranks of the Perlmutter system at NERSC.[2] For this test, we used the CPU-only nodes and did not access the scratch file system in order to give a more reliable estimate of the expected I/O time on typical computing clusters and HPC machines. The total amount of data read during the test was 128.85 gigabytes, and the time spent in I/O operations was less than 5% of the run-time. The POSIX-level data rate was 103.44 gigabytes/s and the MPI-IO-level data rate was 14.43 gigabytes/s. Figure 2 shows that the I/O operations in our improved code are spread evenly over the run-time of the simulation, leading to more file access operations, but smaller data transfers per operation. While the file system would support larger transfer rates, storing the data for processing in the simulation program would require larger RAM allocations, leading to slower overall execution times. This effect becomes particularly

[2]Perlmutter is a Cray Shasta system, using AMD "Milan" EPYC CPUs, a novel Hewlett Packard Enterprise Slingshot high-speed network, and a 35-petabyte FLASH scratch file system. In total, it is composed of 3072 CPU-only and 1792 GPU-accelerated nodes. https://nersc.gov/systems/perlmutter.

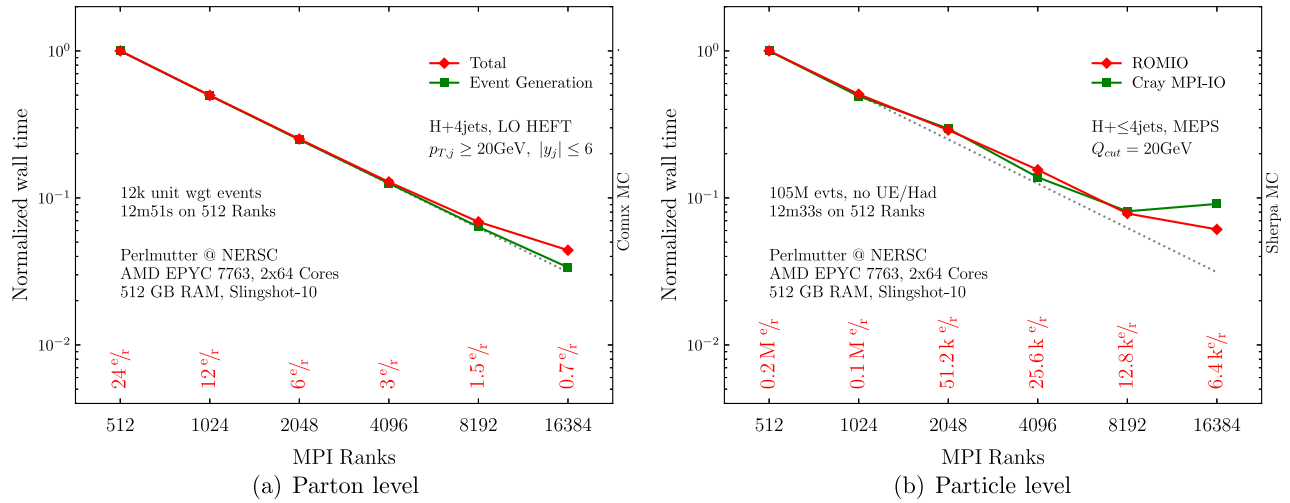


FIG. 3. Strong scaling test of the simulations. (a) Parton level. (b) Particle level. At particle level, we do not include the simulation of multiple interactions and hadronization, and we do not process events further. All times are normalized to the individual results obtained on 512 MPI ranks.

important at larger scales, of the order of 1000 ranks and beyond, where the aggregate time needed for heap allocation would constitute a substantial part of the total run-time and break the strong scaling. This concluded our optimization of the particle-level simulation.

Figure 3 shows the strong scaling tests for the parton- and the particle-level component of the simulation. The test is performed on the Perlmutter system at NERSC. We begin at a scale similar to the upper end of the tested range in [54]. For the parton-level calculations presented in Fig. 3(a), we observe good scaling properties up to 8192 MPI ranks. The green line shows the time spent in the event generation, while the red line displays the time spent for the overall simulation, including initialization and I/O. The numbers at the bottom of the plot show the number of unweighted events generated per MPI rank. We note that the work for this test was selected such that the minimum run-time would correspond to about 30 s, below which the initialization time of the executable consumes a significant fraction of the overall run-time. In practice, one would rather choose run times that are significantly longer, in order to minimize the impact of initialization. At the particle level, shown in Fig. 3(b), we observe scaling up to about 1024 MPI ranks, above which the behavior depends on the particular implementation of MPI-IO. The green line shows the results obtained with the proprietary Cray implementation [81] of MPI-IO, and the red line shows the results obtained with the ROMIO implementation [82]. The numbers at the bottom of the plot show the number of events generated per MPI rank. Above 16384 ranks, the Cray implementation suffers from a problem that prevents collective open calls through HDF5. The ROMIO implementation does allow collective open calls, but does not reach the full performance of the Cray MPI-IO library in data transfer. However, we note that at this scale only about 6400 events are processed per rank, leading to an overall run-time of about 1 min for a total of 105×10^6 events. There is no practical need to perform a calculation of this scale in less than an hour; therefore, our example should be seen as a test of the absolute limits of the code. We believe that further optimization is not needed at this stage. In addition, we note that the particle-level simulation was limited to the perturbative event phases, i.e., we did not include hadronization, underlying event simulation, and hadron decays. Because of the reduced event processing time in this scenario, any scaling violations observed in our test are more severe than in practical applications.

We would like to conclude this section with a seemingly obvious but practically very important remark on the limits of scalability. The aim of an efficient parallel code is to maximize the effective computation time per worker node, i.e., the time spent in useful computations between I/O operations, with the I/O contributing an insignificant fraction of the overall run-time. One of the main reasons for scaling violations to occur is that the time between I/O operations becomes too short because of the limited size of input files. This can lead to significant problems at very large scales, where the input files must then be tens or hundreds of gigabytes in size. Therefore, it is not practical for us to attempt scaling tests for particle-level simulations that go beyond $\mathcal{O}(10^4)$ MPI ranks. We note that this intermediate scale parallelism is actually advantageous, because it allows one to access backfill queues at large computing centers.

III. COMPARISON OF PARTON-LEVEL EVENT GENERATORS

In order to make our new event generation framework as versatile and efficient as possible, we include three different parton-level event generators: Amegic [63], Comix [14], and

TABLE II. Benchmarks for the production and HDF5 write out of $pp \to Z + \text{jet}$ events, comparing SHERPA's Comix with Pepper + Chili, on a single core of an Intel Core i3-8300 CPU at 3.70 GHz and 8 megabyte L3 cache. Event samples are generated with a given target for the total cross section uncertainty ("Tot. unc."). "Speedup" gives the wall time gain factor of Pepper + Chili vs SHERPA (Comix). For Pepper + Chili, the lower multiplicities $Z + 0\text{j}$ and $Z + 1\text{j}$ are generated using helicity summing, while the higher ones are generated using helicity sampling, in order to achieve the best possible performance in each case.

		SHERPA (Comix)			Pepper + Chili			
Process	Tot. unc. (%)	Wall time (s)	Memory (USS) (megabytes)	Efficiency (%)	Wall time (s)	Memory (USS) (megabytes)	Efficiency (%)	Speedup
$Z + 0\text{j}$	0.089	68	62	22	10	40	43	6.8
$Z + 1\text{j}$	0.19	76	66	5.3	31	33	10	2.5
$Z + 2\text{j}$	0.99	92	64	0.28	10	35	1.4	9.2
$Z + 3\text{j}$	3.8	95	65	0.037	36	43	0.097	2.6
$Z + 4\text{j}$	14	122	115	0.0050	71	133	0.016	1.7

Pepper [64,65,83]. Amegic and Comix have been workhorses for the LHC community for more than a decade. Details on their construction and performance can be found in the original publications. Pepper is a new matrix-element generator, previously called BlockGen. It is developed as a portable code for standard-candle processes, which currently include V + jets, $t\bar{t}$ + jets, and pure jet production at the tree level. Parallelized execution on accelerators like GPUs is supported in Pepper, although here we make use only of the single-core CPU version.

Table II shows benchmarks for the production of parton-level HDF5 event files on a single CPU thread for the $pp \to Z + n$ jets process with $n = 0, \ldots, 4$ for a given uncertainty target of the total cross section, comparing SHERPA's Comix generator with Pepper, where the latter uses Chili [13] for the phase-space sampling. Prior to event generation, the different phase-space generators are optimized until a given accuracy target is reached to ensure a fair comparison. The benchmark metrics are the wall time for the generation of the event sample, the memory consumption in terms of the applications' unique set size (USS) in RAM, and the fraction of the number of nonzero events over the total number of events generated, i.e., the measured combined efficiency of the phase-space sampling and unweighting ("Eff."). For Pepper + Chili, we switch from using helicity summing for the $n = 0$, 1 multiplicities to using helicity sampling for the $n = 2$, 3, 4 ones, in order to achieve the best performance. We find that the wall times are significantly lower for Pepper + Chili for the given multiplicities, with the speedup factor ranging between 2 and 10. For the higher multiplicities, $n = 3$, 4, the speedup becomes smaller, but is still significant with factors of 2.6 and 1.7, respectively.

Figure 4 shows a cross-check of differential distributions of k_T jet rates between Pepper and Comix, after leading-order multijet merging [37] with SHERPA2.2 [62]. The first ratio panel compares the predictions obtained with PEPPER+SHERPA to the results from COMIX+SHERPA, normalized to the statistical

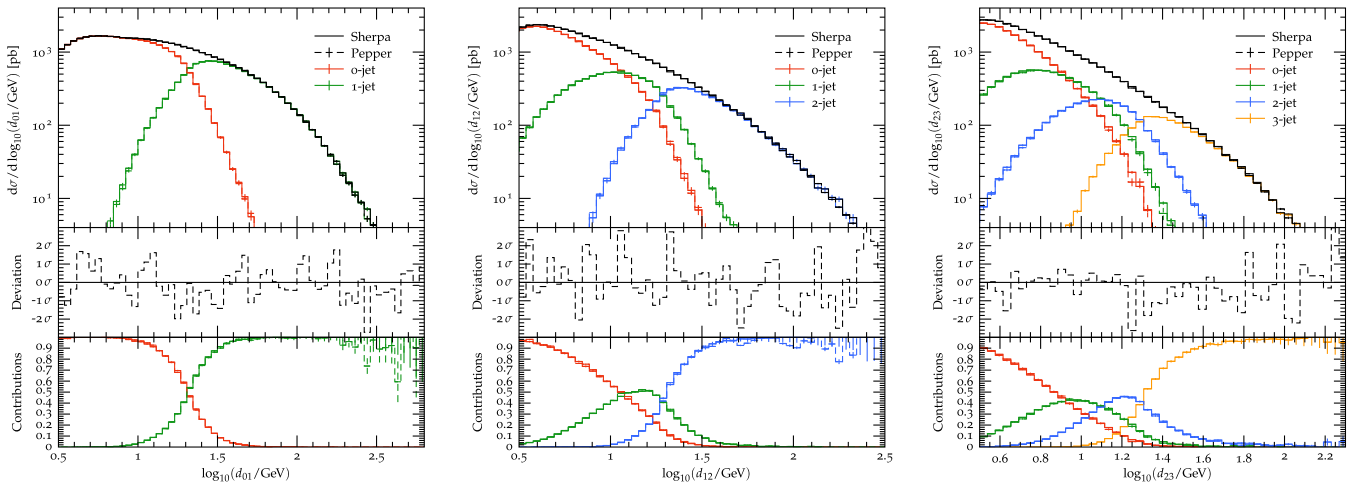


FIG. 4. Differential jet rates for the leading, subleading, and subsubleading jet clustering in $Z + \text{jet}$ production at the LHC. Simulations have been performed with up to one-, two-, and three-jet matrix elements at leading-order QCD. The colored lines represent the contributions from the parton-level inputs with the specified multiplicity.

uncertainty of the latter. The second ratio panel shows the relative contributions from the event samples with $Z+0$ jets, $Z+1$ jet, $Z+2$ jets, and $Z+3$ jets to the overall prediction. It can be seen that the results are in agreement up to statistical fluctuations, which are typically at or below the 1σ level, as expected.

The computational complexity of the event generation techniques used in Pepper scales factorially and will eventually cause Pepper to become slower (and more memory consuming) than Comix at high multiplicity, because Comix uses an algorithm with overall exponential scaling. This exemplifies how our new event generation framework can be used to take advantage of the best possible solution to produce events at parton level: At low to medium multiplicity, Pepper can be used at great efficiency and speed. At high multiplicity, Comix can be used due to the improved scaling. At NLO QCD precision, Amegic can be used for the Born-like components of the calculation, and Comix can be used for the subtracted real-emission terms. In this manner, the total computing time can be reduced to the absolute minimum required for event simulation at today's state of event generator development, in order to provide the experiments with the best possible simulations for their analyses.

IV. COMPARISON OF PARTICLE-LEVEL EVENT GENERATORS

The systematic assessment of uncertainties in particle-level simulations has been vital for the success of the LHC physics program. It is particularly important in cases where the uncertainty is not of parametric type, such as when switching between two formally equivalent, but practically different, NLO QCD matching schemes [29]. When used correctly, the residual variations between event generator predictions give the best possible nonparametric estimate of perturbative (and nonperturbative) uncertainties in the simulation. Our new event generation framework allows one to obtain such uncertainty estimates based on the same parton-level input configurations and at minimal computational cost. This contributes to creating a sustainable computing model for high-energy physics research.

Here we present a simple example of this type for the standard-candle process of Z-boson plus multijet production. We consider proton-proton collisions at the high-luminosity LHC at $\sqrt{s} = 14$ TeV. The complete setup has been described in [54]. In particular, we use the CT14 next-to-next-to-leading order (NNLO) PDF set [84] and define the strong coupling accordingly. Our modified parton-level event generator is based on Comix [14] as included in SHERPA version 2.2.4 [61,62]. Our modified particle-level event generators are based on PYTHIA 8 [66] and SHERPA2.2 [62], including the improvements reported in [85,86]. Jets are defined using the k_T clustering algorithm with $R = 0.4$, $p_{T,j} > 20$ GeV, and $|\eta_j| < 6$. Following the good agreement between parton- and particle-level results established in [87,88], and the good agreement between fixed-order and MINLO [89] results established in [90,91], the renormalization and factorization scales are set to $\hat{H}'_T/2$, where $\hat{H}'_T = \sum_{j\in\text{jets}} p_{t,j} + \sqrt{m^2_{l\bar{l}} + p^2_{T,l\bar{l}}}$.

Figure 5 shows the transverse momentum spectra of the leading, subleading, and subsubleading jet in the simulation. The colored lines correspond to the contributions from the individual parton-level input samples after the full simulation. The upper ratio panel shows the ratio between the SHERPA and the PYTHIA predictions. This ratio is of the order of 10%, which can be ascribed to differences in the parton-shower algorithm used in the two different generators. This uncertainty should be added as a variation to the parametric scale uncertainties, which we investigate in Sec. V.

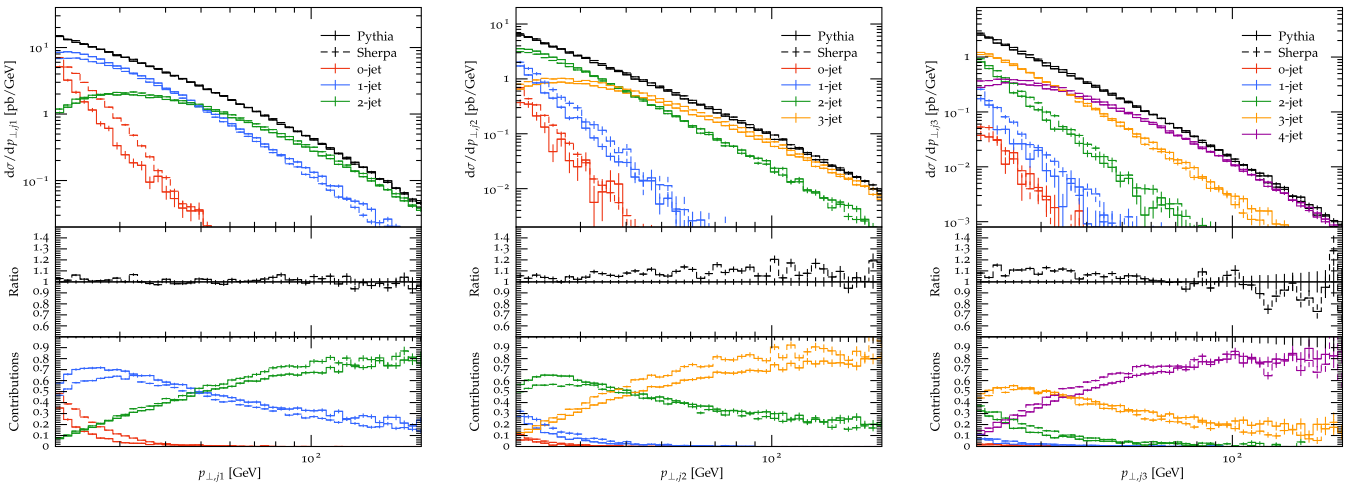


FIG. 5. Transverse momentum spectra of the leading, subleading, and subsubleading jet in $Z+$jets production at the LHC. Simulations have been performed with up to 2-, 3-, and 4-jet matrix elements at leading-order QCD. The colored lines represent the contributions from the parton-level inputs with the specified multiplicity.

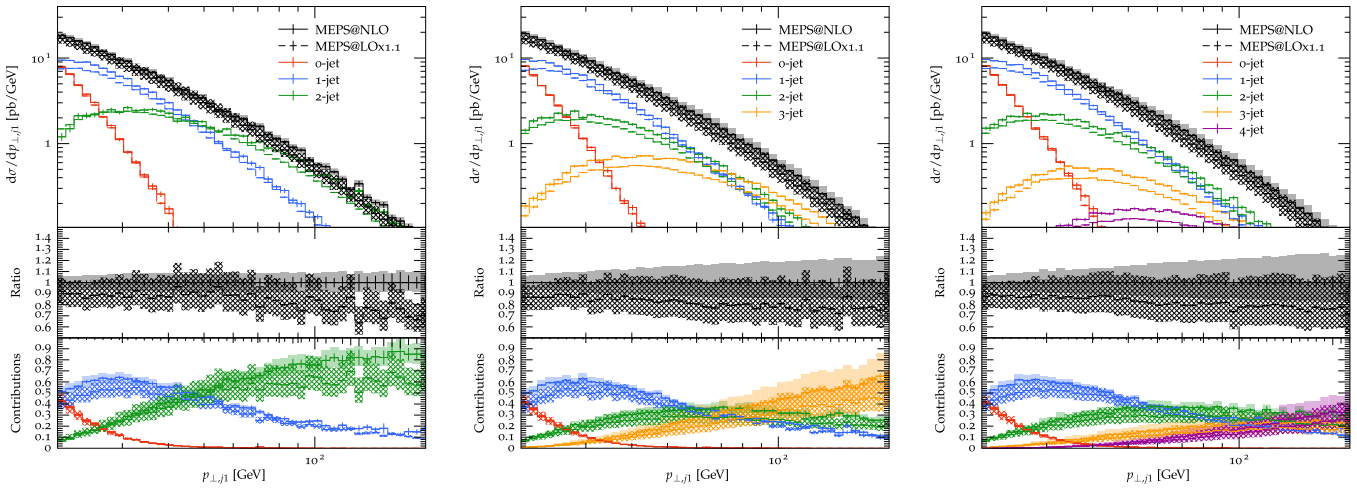


FIG. 6. Transverse momentum spectrum of the leading jet in Z + jets production at the LHC, simulated using multijet merging for up to two jets at NLO with up to zero, one, and two additional jets at leading-order precision (from left to right), compared to a purely LO multijet merged prediction with the same overall multiplicities. The colored lines represent the contributions from the parton-level inputs with the specified multiplicity, and the hatched and solid bands indicate the uncertainties from renormalization and factorization scale variations at leading and next-to-leading order.

Figure 6 shows the transverse momentum spectrum of the leading jet in a multijet merged setup with up to two jets computed at next-to-leading-order precision and with up to zero, one, and two additional jets computed at leading-order precision [92]. For reference, we also show the prediction from a leading-order multijet merged event sample with identical jet multiplicity (dashed lines). The leading-order predictions have been scaled such as to reproduce the total cross section of the next-to-leading-order predictions. The colored lines correspond to the contributions from the individual parton-level input samples after the full simulation. The hatched bands indicate the scale uncertainties from a seven-point scale variation at leading order, and the solid bands represent the corresponding uncertainties at next-to-leading-order precision. Note that the scale uncertainties increase with increasing jet multiplicity in the merging. This is an artifact of the method to estimate the scale uncertainty in the complete calculation and is due to the fact that scales are varied in the computation of the hard matrix elements alone. It also indicates the importance of higher-multiplicity final states for the experimental observable. To obtain a comprehensive picture of the uncertainty, the renormalization scale-dependent terms of the parton-shower resummation at higher-logarithmic order should be taken into account. This is the topic of active research elsewhere [93,94], and we will therefore not discuss the effect in this publication. We emphasize, however, that the simulation of additional radiation at tree level is necessary for a proper physics modeling of high-multiplicity final states, and it is therefore not sufficient to limit the fixed-order perturbative calculations to low multiplicity. This is where the increased efficiency of our event generation framework becomes relevant for practical applications at the LHC.

We have validated the framework presented in this paper using the ATLAS benchmark setups described in [85]. The event files can be found at [95,96]. For practical applications where multiple particle-level simulations are generated with the same parton-level input, the LHEH5 event file technology will result in a significantly reduced overall production cost. There may, however, be remaining obstacles to implementing the method in large-scale event production for the LHC experiments, in particular, the access of subsamples and the synchronization of subsamples across various sites of the Worldwide LHC Computing Grid (WLCG). The solution to this problem must be found in collaboration with experts from the LHC experiments, who are proficient in WLCG workflows. We therefore postpone the discussion to a future publication.

V. HIGGS-BOSON PLUS MULTIJET PRODUCTION AS AN EXAMPLE APPLICATION

With an anticipated 3 ab^{-1} at the high-luminosity LHC, Higgs-boson plus multijet events will be copiously produced, and even the six-jet final state will be measurable at good precision. While not a discovery channel in its own right, the Higgs-boson plus multijet signature can be used to test the dynamics of the Standard Model, and it also provides the background to a number of Higgs-boson related measurements and searches, such as di-Higgs production. In anticipation of these analyses, it behooves us to provide precision simulations. In this subsection, we therefore present the first study of Higgs-boson production through gluon fusion at the LHC, with up to seven additional jets computed at LO QCD and up to two jets computed at NLO QCD in the Higgs effective theory [97,98]. We use the MEPS@NLO algorithm [40,41] to merge these calculations

into an inclusive event sample. The parton-level inputs are generated using Amegic [63], Comix [14], and MCFM [99–105]

We consider proton-proton collisions at the high-luminosity LHC at $\sqrt{s} = 14$ TeV. The basic setup has been described in [54]. We use the CT14 NNLO PDF set [84] and define the strong coupling accordingly. Our modified parton-level event generator is based on Comix [14] as included in SHERPA version 2.2.4 [61,62]. Our modified particle-level event generator is based on SHERPA version 2.2 [61,62]. Jets are defined using the k_T clustering algorithm with $R = 0.4$, $p_{T,j} > 20$ GeV, and $|\eta_j| < 6$. Following the good agreement between parton- and particle-level results established in [87], the renormalization and factorization scales are set to $\hat{H}_{T,m}/2$, where $\hat{H}_{T,m} = \sum_{j \in \text{jets}} p_{t,j} + \sqrt{m_H^2 + p_{T,H}^2}$.

Figure 7 shows the jet transverse momentum spectra at leading order (top panels) and at next-to-leading order (bottom panels). We compare multijet merged simulations where the maximum jet multiplicity $n_{\max}$ is set to the

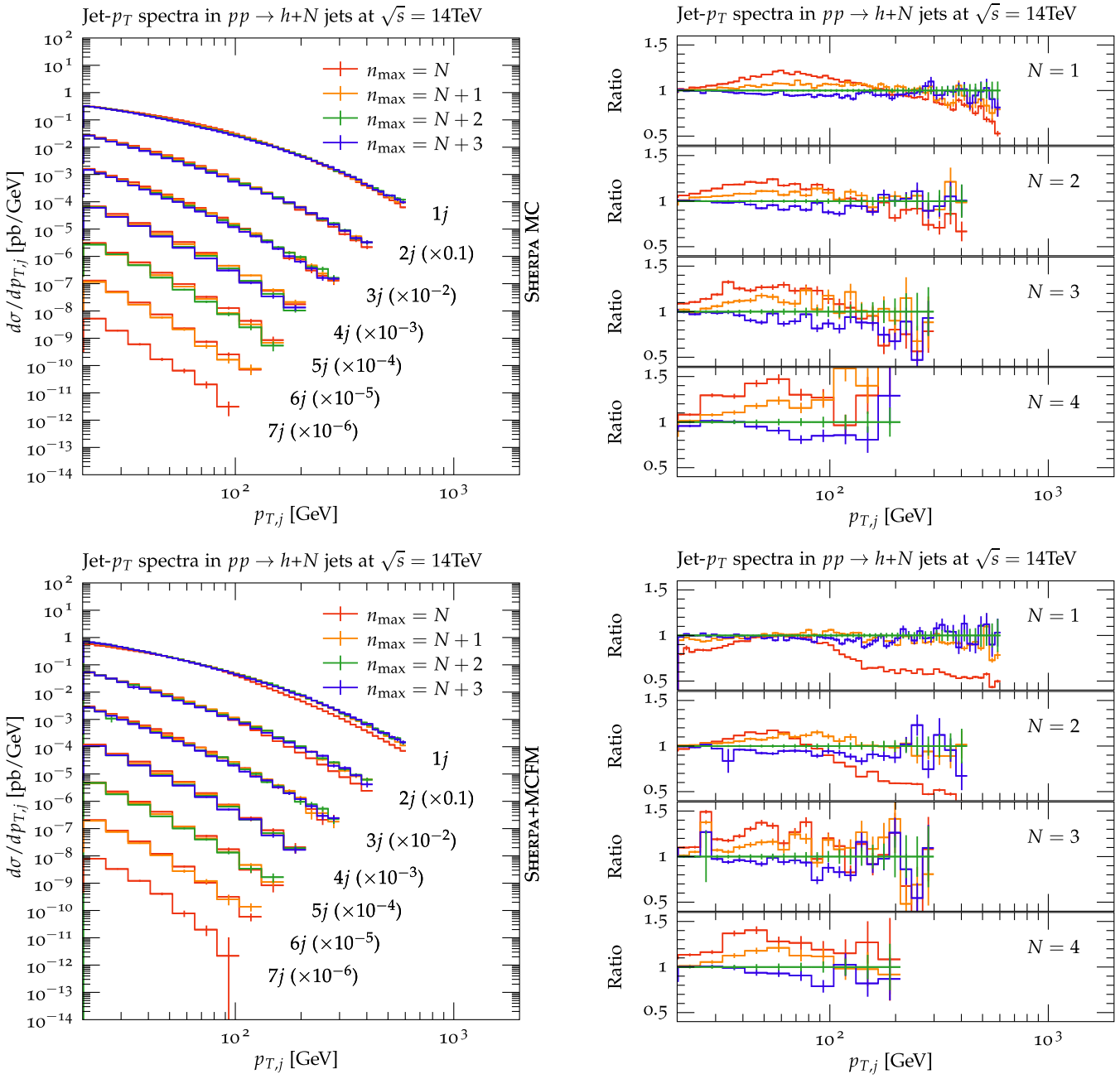


FIG. 7. Jet transverse momentum distributions in Higgs-boson + jets events, computed using multijet merging with maximum jet multiplicity equal to N (red), $N+1$ (green), $N+2$ (blue), and $N+3$ (purple), with N the number of measured jets. The top panels show leading order, the bottom panels show next-to-leading-order merged results with $n_{\text{max,NLO}} = 2$.

number of measured jets, N (red), to $N+1$ (green), $N+2$ (blue), and $N+3$ (purple). For NLO merged simulations, the maximum number of jets computed at NLO precision is $n_{\mathrm{max,NLO}}=2$, and we apply local K factors based on this calculation to higher jet multiplicities. The panels on the right show the ratio between different predictions, normalized to the result for $n_{\mathrm{max}}=N+2$. It can be seen that the NLO merged predictions are more stable with respect to variations of n_{max} at $N=1$, as expected from the higher precision of the calculation at low jet multiplicity. At $N=2$, this effect is diluted by higher-multiplicity tree-level contributions, as explained in Sec. IV and Fig. 6. NLO accurate predictions for three jets at parton level would help to alleviate this problem [106,107]. However, we could not generate the corresponding unweighted event samples within the limited computing budget for this publication, and we therefore leave a detailed investigation to future work.

VI. CONCLUSIONS

We have presented a new framework for the precise and efficient simulation of events in collider experiments, with particular emphasis on the high-luminosity Large Hadron Collider. The new technique is especially suited for the physics modeling of high-multiplicity final states as it allows one to match parton-level calculations at next-to-leading-order QCD precision to parton showers and merge multiple exclusive calculations into inclusive predictions. Parametric uncertainty estimates can be computed on the fly, using the techniques from [108]. There are no restrictions on the variations that can be performed, and the variations do not need to be included at the time of parton-level event production. We have demonstrated scalability of our approach on a state of the art high-performance computer at a leadership class computing facility. With the computing demands of the LHC experiments becoming an ever more pressing problem due to increased precision in the measurements, our new framework presents an important step toward a more flexible as well as economically and ecologically sustainable approach to event generation in the high-luminosity era. We have validated the new technology against previous simulation programs and enabled event production with a modern, portable parton-level event generator.

ACKNOWLEDGMENTS

We are grateful to Rui Wang for collaboration in the initial stages of the project. We would like to thank Christian Preuss for his help in porting the HDF5 technology to the most recent version of the PYTHIA event generator. This research was supported by the Fermi National Accelerator Laboratory (Fermilab), a U.S. Department of Energy, Office of Science, HEP User Facility. Fermilab is managed by Fermi Research Alliance, LLC (FRA), acting under Contract No. DE–AC02–07CH11359. The work of T. C., S. H., P. H., J. I., and R. L. was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program, grant "HPC framework for event generation at colliders." E. B. and M. K. acknowledge support from BMBF (Contract No. 05H21MGCAB). Their research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—456104544; 510810461. C. G. is supported by the STFC SoftWare and InFrastructure Technology for High Energy Physics project (Grant No. ST/V002627/1). This research used the Fermilab Wilson Institutional Cluster for code development, testing, and validation. We are grateful to James Simone for his support.

[1] J. M. Campbell *et al.*, Event generators for high-energy physics experiments, in *2022 Snowmass Summer Study* (2022), arXiv:2203.11110.
[2] S. Amoroso *et al.* (HSF Physics Event Generator WG Collaboration), Challenges in Monte Carlo event generator software for high-luminosity LHC, Comput. Software Big Sci. **5**, 12 (2021).
[3] E. Yazgan *et al.* (HSF Physics Event Generator WG Collaboration), HL-LHC computing review stage-2, common software projects: Event generators, arXiv:2109.14938.
[4] F. James, Monte-Carlo phase space, Report No. CERN-68-15, 1968.
[5] E. Byckling and K. Kajantie, Reductions of the phase-space integral in terms of simpler processes, Phys. Rev. **187**, 2008 (1969).
[6] E. Byckling and K. Kajantie, N-particle phase space in terms of invariant momentum transfers, Nucl. Phys. **B9**, 568 (1969).
[7] R. Kleiss, W. J. Stirling, and S. D. Ellis, A new Monte Carlo treatment of multiparticle phase space at high energies, Comput. Phys. Commun. **40**, 359 (1986).
[8] C. G. Papadopoulos, PHEGAS: A phase-space generator for automatic cross-section computation, Comput. Phys. Commun. **137**, 247 (2001).
[9] F. Maltoni and T. Stelzer, MadEvent: Automatic event generation with MadGraph, J. High Energy Phys. 02 (2002) 027.
[10] A. van Hameren and C. G. Papadopoulos, A hierarchical phase space generator for QCD antenna structures, Eur. Phys. J. C **25**, 563 (2002).

[11] A. van Hameren, Kaleu: A general-purpose parton-level phase space generator, arXiv:1003.4953.
[12] S. Plätzer, RAMBO on diet, arXiv:1308.2922.
[13] E. Bothmann, T. Childers, W. Giele, F. Herren, S. Höche, J. Isaacsson, M. Knobbe, and R. Wang, Efficient phase-space generation for hadron collider event simulation, SciPost Phys. **15**, 169 (2023).
[14] T. Gleisberg and S. Höche, Comix, A new matrix element generator, J. High Energy Phys. 12 (2008) 039.
[15] F. A. Berends, R. Kleiss, P. De Causmaecker, R. Gastmans, and T. T. Wu, Single bremsstrahlung processes in gauge theories, Phys. Lett. **103B**, 124 (1981).
[16] F. A. Berends and W. Giele, The six-gluon process as an example of Weyl-van der Waerden spinor calculus, Nucl. Phys. **B294**, 700 (1987).
[17] M. L. Mangano, S. J. Parke, and Z. Xu, Duality and multi-gluon scattering, Nucl. Phys. **B298**, 653 (1988).
[18] F. A. Berends and W. T. Giele, Recursive calculations for processes with n gluons, Nucl. Phys. **B306**, 759 (1988).
[19] F. A. Berends, W. T. Giele, and H. Kuijf, Exact expressions for processes involving a vector boson and up to five partons, Nucl. Phys. **B321**, 39 (1989).
[20] A. Kanaki and C. G. Papadopoulos, HELAC: A package to compute electroweak helicity amplitudes, Comput. Phys. Commun. **132**, 306 (2000).
[21] M. L. Mangano, M. Moretti, F. Piccinini, R. Pittau, and A. D. Polosa, ALPGEN, A generator for hard multiparton processes in hadronic collisions, J. High Energy Phys. 07 (2002) 001.
[22] A. Cafarella, C. G. Papadopoulos, and M. Worek, HELAC-PHEGAS: A generator for all parton level processes, Comput. Phys. Commun. **180**, 1941 (2009).
[23] A. Lifson and O. Mattelaer, Improving colour computations in MadGraph5_aMC@NLO and exploring a $1/N_c$ expansion, Eur. Phys. J. C **82**, 1144 (2022).
[24] B. Webber, Monte Carlo simulation of hard hadronic processes, Annu. Rev. Nucl. Part. Sci. **36**, 253 (1986).
[25] A. Buckley *et al.*, General-purpose event generators for LHC physics, Phys. Rep. **504**, 145 (2011).
[26] S. Frixione and B. R. Webber, Matching NLO QCD computations and parton shower simulations, J. High Energy Phys. 06 (2002) 029.
[27] P. Nason, A new method for combining NLO QCD with shower Monte Carlo algorithms, J. High Energy Phys. 11 (2004) 040.
[28] S. Frixione, P. Nason, and C. Oleari, Matching NLO QCD computations with parton shower simulations: The POWHEG method, J. High Energy Phys. 11 (2007) 070.
[29] S. Höche, F. Krauss, M. Schönherr, and F. Siegert, A critical appraisal of NLO + PS matching methods, J. High Energy Phys. 09 (2011) 049.
[30] S. Plätzer and S. Gieseke, Dipole showers and automated NLO matching in Herwig++, Eur. Phys. J. C **72**, 2187 (2012).
[31] S. Catani, F. Krauss, R. Kuhn, and B. R. Webber, QCD matrix elements + parton showers, J. High Energy Phys. 11 (2001) 063.
[32] M. L. Mangano, M. Moretti, and R. Pittau, Multijet matrix elements and shower evolution in hadronic collisions: $Wb\bar{b} + n$-jets as a case study, Nucl. Phys. **B632**, 343 (2002).
[33] F. Krauss, Matrix elements and parton showers in hadronic interactions, J. High Energy Phys. 08 (2002) 015.
[34] L. Lönnblad, Correcting the colour-dipole cascade model with fixed order matrix elements, J. High Energy Phys. 05 (2001) 046.
[35] N. Lavesson and L. Lönnblad, Merging parton showers and matrix elements—back to basics, J. High Energy Phys. 04 (2007) 085.
[36] J. Alwall *et al.*, Comparative study of various algorithms for the merging of parton showers and matrix elements in hadronic collisions, Eur. Phys. J. C **53**, 473 (2008).
[37] S. Höche, F. Krauss, S. Schumann, and F. Siegert, QCD matrix elements and truncated showers, J. High Energy Phys. 05 (2009) 053.
[38] K. Hamilton, P. Richardson, and J. Tully, A modified CKKW matrix element merging approach to angular-ordered parton showers, J. High Energy Phys. 11 (2009) 038.
[39] L. Lönnblad and S. Prestel, Matching tree-level matrix elements with interleaved showers, J. High Energy Phys. 03 (2011) 019.
[40] S. Höche, F. Krauss, M. Schönherr, and F. Siegert, QCD matrix elements + parton showers: The NLO case, J. High Energy Phys. 04 (2013) 027.
[41] T. Gehrmann, S. Höche, F. Krauss, M. Schönherr, and F. Siegert, NLO QCD matrix elements + parton showers in $e^+e^- \to$ hadrons, J. High Energy Phys. 01 (2012) 144.
[42] L. Lönnblad and S. Prestel, Merging multi-leg NLO matrix elements with parton showers, J. High Energy Phys. 03 (2013) 166.
[43] R. Frederix and S. Frixione, Merging meets matching in MC@NLO, J. High Energy Phys. 12 (2012) 061.
[44] J. Bellm, S. Gieseke, and S. Plätzer, Merging NLO multi-jet calculations with improved unitarization, Eur. Phys. J. C **78**, 244 (2018).
[45] H.-U. Bengtsson and T. Sjöstrand, The Lund Monte Carlo for hadronic processes: PYTHIA version 4.8, Comput. Phys. Commun. **46**, 43 (1987).
[46] B. Andersson, *The Lund Model* (Cambridge University Press, Cambridge, England, 2005), Vol. 7.
[47] T. D. Gottschalk, A realistic model for e^+e^- annihilation including parton bremsstrahlung effects, Nucl. Phys. **B214**, 201 (1983).
[48] T. D. Gottschalk, An improved description of hadronization in the QCD cluster model for e^+e^- annihilation, Nucl. Phys. **B239**, 349 (1984).
[49] B. R. Webber, A QCD model for jet fragmentation including soft gluon interference, Nucl. Phys. **B238**, 492 (1984).
[50] J.-C. Winter, F. Krauss, and G. Soff, A modified cluster-hadronisation model, Eur. Phys. J. C **36**, 381 (2004).
[51] T. Sjöstrand and M. van Zijl, A multiple-interaction model for the event structure in hadron collisions, Phys. Rev. D **36**, 2019 (1987).
[52] T. Sjöstrand and P. Z. Skands, Transverse-momentum-ordered showers and interleaved multiple interactions, Eur. Phys. J. C **39**, 129 (2005).
[53] M. Bähr, J. M. Butterworth, and M. H. Seymour, The underlying event and the total cross section from tevatron to the LHC, J. High Energy Phys. 01 (2008) 065.

[54] S. Höche, S. Prestel, and H. Schulz, Simulation of vector boson plus many jet final states at the high luminosity LHC, Phys. Rev. D **100**, 014024 (2019).

[55] H. Brooks and C. T. Preuss, Efficient multi-jet merging with the Vincia sector shower, Comput. Phys. Commun. **264**, 107985 (2021).

[56] T. Sjöstrand, S. Mrenna, and P. Skands, PYTHIA 6.4 physics and manual, J. High Energy Phys. 05 (2006) 026.

[57] J. Alwall *et al.*, A standard format for Les Houches event files, Comput. Phys. Commun. **176**, 300 (2007).

[58] J. R.Andersen *et al.*, Les Houches 2017: Physics at TeV colliders standard model working group report, arXiv: 1803.07977.

[59] E. Bothmann *et al.*, A standard convention for particle-level Monte Carlo event-variation weights, SciPost Phys. Core **6**, 007 (2023).

[60] J. T. Childers, T. D. Uram, T. J. LeCompte, M. E. Papka, and D. P. Benjamin, Adapting the serial Alpgen parton-interaction generator to simulate LHC collisions on millions of parallel threads, Comput. Phys. Commun. **210**, 54 (2017).

[61] T. Gleisberg, S. Höche, F. Krauss, M. Schönherr, S. Schumann, F. Siegert, and J. Winter, Event generation with SHERPA1.1, J. High Energy Phys. 02 (2008) 007.

[62] E. Bothmann *et al.* (Sherpa Collaboration), Event generation with SHERPA2.2, SciPost Phys. **7**, 034 (2019).

[63] F. Krauss, R. Kuhn, and G. Soff, Amegic++ 1.0: A matrix element generator in C++, J. High Energy Phys. 02 (2001) 044.

[64] E. Bothmann, W. Giele, S. Höche, J. Isaacson, and M. Knobbe, Many-gluon tree amplitudes on modern GPUs: A case study for novel event generators, SciPost Phys. Codebases 3 (2022), 10.21468/SciPostPhysCodeb.3.

[65] E. Bothmann, T. Childers, W. Giele, S. Höche, J. Isaacson, and M. Knobbe, A portable parton-level event generator for the high-luminosity LHC, arXiv:2311.06198.

[66] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, An introduction to PYTHIA 8.2, Comput. Phys. Commun. **191**, 159 (2015).

[67] 10.5281/zenodo.7751000.

[68] 10.5281/zenodo.7747376.

[69] https://gitlab.com/hpcgen/.

[70] https://gitlab.com/sherpa-team/sherpa/-/tree/rel-2-3-0.

[71] S. Frixione, Z. Kunszt, and A. Signer, Three-jet cross-sections to next-to-leading order, Nucl. Phys. **B467**, 399 (1996).

[72] S. Catani and M. H. Seymour, A general algorithm for calculating jet cross sections in NLO QCD, Nucl. Phys. **B485**, 291 (1997).

[73] S. Catani, S. Dittmaier, M. H. Seymour, and Z. Trocsanyi, The dipole formalism for next-to-leading order QCD calculations with massive partons, Nucl. Phys. **B627**, 189 (2002).

[74] Together with this publication, we provide a set of simple tools to parse event files written in the new format, to merge two event files, and to filter event files for overweight and zero weight events. The source code can be found at https://gitlab.com/hpcgen/tools.

[75] The HDF Group, Hierarchical data format, version 5 (1997–2023), https://www.hdfgroup.org/HDF5/.

[76] The MPI forum, MPI-2: Extensions to the message-passing interface (1997), http://www.mpi-forum.org/docs/docs.html.

[77] HighFive—HDF5 header-only $C++$ library, https://bluebrain.github.io/HighFive/.

[78] B. Behzad, H. V. T. Luu, J. Huchette, S. Byna, Prabhat, R. Aydt, Q. Koziol, and M. Snir, Taming parallel i/o complexity with auto-tuning, in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, SC '13 (Association for Computing Machinery, New York, 2013).

[79] P. Carns, R. Latham, R. Ross, K. Iskra, S. Lang, and K. Riley, 24/7 characterization of petascale I/O workloads, in *Proceedings of 2009 Workshop on Interfaces and Architectures for Scientific Data Storage* (2009).

[80] P. Carns, K. Harms, W. Allcock, C. Bacon, S. Lang, R. Latham, and R. Ross, Understanding and improving computational science storage access through continuous characterization, ACM Trans. Storage **7**, 1 (2011).

[81] Cray MPICH documentation, https://cpe.ext.hpe.com/docs/mpt/mpich/.

[82] ROMIO documentation, https://wordpress.cels.anl.gov/romio/.

[83] E. Bothmann, J. Isaacson, M. Knobbe, S. Höche, and W. Giele, QCD tree amplitudes on modern GPUs: A case study for novel event generators, Proc. Sci. ICHEP2022 (**2022**) 222.

[84] S. Dulat, T.-J. Hou, J. Gao, M. Guzzi, J. Huston, P. Nadolsky, J. Pumplin, C. Schmidt, D. Stump, and C. P. Yuan, New parton distribution functions from a global analysis of quantum chromodynamics, Phys. Rev. D **93**, 033006 (2016).

[85] E. Bothmann, A. Buckley, I. A. Christidi, C. Gütschow, S. Höche, M. Knobbe, T. Martin, and M. Schönherr, Accelerating LHC event generation with simplified pilot runs and fast PDFs, Eur. Phys. J. C **82**, 1128 (2022).

[86] K. Danziger, S. Höche, and F. Siegert, Reducing negative weights in Monte Carlo event generation with SHERPA, arXiv:2110.15211.

[87] J. Bellm *et al.*, Jet cross sections at the LHC and the quest for higher precision, Eur. Phys. J. C **80**, 93 (2020).

[88] A. Buckley *et al.*, A comparative study of Higgs boson production from vector-boson fusion, J. High Energy Phys. 11 (2021) 108.

[89] K. Hamilton, P. Nason, and G. Zanderighi, MINLO: Multi-scale improved NLO, J. High Energy Phys. 10 (2012) 155.

[90] S. Höche, P. Maierhöfer, N. Moretti, S. Pozzorini, and F. Siegert, Next-to-leading order QCD predictions for top-quark pair production with up to three jets, Eur. Phys. J. C **77**, 145 (2017).

[91] F. R. Anger, F. Febres Cordero, S. Höche, and D. Maître, Weak vector boson production with many jets at the LHC $\sqrt{s} = 13$ TeV, Phys. Rev. D **97**, 096010 (2018).

[92] 10.5281/zenodo.8226865.

[93] F. Dulat, S. Höche, and S. Prestel, Leading-color fully differential two-loop soft corrections to QCD dipole showers, Phys. Rev. D **98**, 074013 (2018).

[94] S. Ferrario Ravasio, K. Hamilton, A. Karlberg, G. P. Salam, L. Scyboz, and G. Soyez, A parton shower with higher-logarithmic accuracy for soft emissions, Phys. Rev. Lett. **131**, 161906 (2023).

[95] 10.5281/zenodo.8298371.
[96] 10.5281/zenodo.8298334.
[97] S. Dawson, Radiative corrections to Higgs boson production, Nucl. Phys. **B359**, 283 (1991).
[98] A. Djouadi, M. Spira, and P. Zerwas, Production of Higgs bosons in proton colliders: QCD corrections, Phys. Lett. B **264**, 440 (1991).
[99] J. M. Campbell and R. K. Ellis, Update on vector boson pair production at hadron colliders, Phys. Rev. D **60**, 113006 (1999).
[100] J. M. Campbell, R. K. Ellis, and G. Zanderighi, Next-to-leading order Higgs +2 jet production via gluon fusion, J. High Energy Phys. 10 (2006) 028.
[101] S. Badger, J. M. Campbell, R. K. Ellis, and C. Williams, Analytic results for the one-loop NMHV Hqqgg amplitude, J. High Energy Phys. 12 (2009) 035.
[102] J. M. Campbell, R. K. Ellis, and C. Williams, Vector boson pair production at the LHC, J. High Energy Phys. 07 (2011) 018.
[103] J. M. Campbell, R. K. Ellis, and W. T. Giele, A multi-threaded version of MCFM, Eur. Phys. J. C **75**, 246 (2015).
[104] J. Campbell and T. Neumann, Precision phenomenology with MCFM, J. High Energy Phys. 12 (2019) 034.
[105] J. M. Campbell, S. Höche, and C. T. Preuss, Accelerating LHC phenomenology with analytic one-loop amplitudes: A C++ interface to MCFM, Eur. Phys. J. C **81**, 1117 (2021).
[106] N. Greiner, S. Höche, G. Luisoni, M. Schönherr, J.-C. Winter, and V. Yundin, Phenomenological analysis of Higgs boson production through gluon fusion in association with jets, J. High Energy Phys. 01 (2015) 169.
[107] N. Greiner, S. Höche, G. Luisoni, M. Schönherr, and J.-C. Winter, Full mass dependence in Higgs boson production in association with jets at the LHC and FCC, J. High Energy Phys. 01 (2016) 091.
[108] E. Bothmann, M. Schönherr, and S. Schumann, Reweighting QCD matrix-element and parton-shower calculations, Eur. Phys. J. C **76**, 590 (2016).