arXiv:1907.09874 [hep-ph]
LU TP 19-31
MCnet-19-16
July 2019

The PYTHIA Event Generator: Past, Present and Future*

Torbjörn Sjöstrand

Theoretical Particle Physics,
Department of Astronomy and Theoretical Physics,
Lund University,
Sölvegatan 14A,
SE-223 62 Lund, Sweden

Abstract

The evolution of the widely-used PYTHIA particle physics event generator is outlined, from the early days to the current status and plans. The key decisions and the development of the major physics components are put in context.

---

[*] submitted to 50 Years of Computer Physics Communications: a special issue focused on computational science software

1 Introduction

The PYTHIA event generator is one of the most commonly used pieces of software in particle physics and related areas, either on its own or "under the hood" of a multitude of other programs. The program is designed to simulate the physics processes that can occur in collisions between high-energy particles, e.g. at the LHC collider at CERN. Monte Carlo methods are used to represent the quantum mechanical variability that can give rise to wildly different multiparticle final states under fixed simple initial conditions. A combination of perturbative results and models for semihard and soft physics — many of them developed in the PYTHIA context — are combined to trace the evolution towards complex final states.

The program roots stretch back over forty years to the JETSET program, with which it later was fused. Twelve manuals for the two programs have been published in Computer Physics Communications (CPC), Table 1, together collecting over 20 000 citations in the Inspire database. A longer (580 pp) physics description and manual, published in JHEP, is one of only six Inspire entries to exceed 10 000 citations. This was an exception, however, and JHEP immediately recognized its "mistake" by thenceforth banning all publication of manuals. Thus CPC has remained the premier place where descriptions of high energy physics programs has always been welcome, which deserves to be recognized at this 50^{th} anniversary of CPC.

In the present contribution to the celebration, I will outline the historical evolution of PYTHIA/JETSET, its current status and plans for the future. The description is anecdotal, with emphasis on some of the early decisions and key concepts that came to shape the continued evolution of the program(s). Therefore it should not be viewed as a full-scale review of the PYTHIA physics and code, let alone of the broader fields of particle physics phenomenology and event generators. The text is also not strictly chronological, but typically presents a topic at a time when it became important, with brief comments on earlier roots and later developments. Finally, since so many diverse topics enter in passing, the bibliography had to be restrictive. Further references can be found in the articles quoted, notably in the PYTHIA 6.4 manual [16] and in some reviews [20, 21, 22].

2 In the beginning

The first code of what was to become JETSET was written in May 1978. It sprung out of studies in Lund on the structure of the fragmentation of a single quark into a jet of hadrons, that had begun about a year earlier, led by Bo Andersson and Gösta Gustafson [23, 24]. These early studies were based on a recursive structure, expressible in terms of an integral equation that could be solved analytically in simple cases. The article by Field and Feynman [25] introduced a similar model, but also took the step of simulating the process on a computer with Monte Carlo techniques. Thereby each quark jet became associated with an explicit list of particles, opening up for more detailed studies than is possible analytically. At the time it was quite a novel idea to most people. Actually, Artru and Mennessier had introduced and simulated a very interesting fragmentation model some years earlier [26], for a kicked-apart quark–antiquark pair connected by a linear-potential force-field "string". With a constant probability per unit space–time area for the string to

Table 1: The main versions of Jetset and Pythia, with their date of appearance and published manuals where relevant.

Jetset versions		
No.	Date	Publ.
1	Nov 78	[1]
2	May 79	[2]
3.1	Aug 79	—
3.2	Apr 80	[3]
3.3	Aug 80	—
4.1	Apr 81	—
4.2	Nov 81	—
4.3 G	Mar 82	[4]
4.3 E	Jul 82	[5]
5.1	Apr 83	—
5.2	Nov 83	—
5.3	May 84	—
6.1	Jan 85	—
6.2	Oct 85	[6]
6.3	Oct 86	[7]
7.1	Feb 89	—
7.2	Nov 89	—
7.3	May 90	[8]
7.4	Dec 93	[9]

Pythia versions		
No.	Date	Publ.
1	Dec 82	[10]
2	—	
3.1	—	
3.2	—	
3.3	Feb 84	[11]
3.4	Sep 84	[11]
4.1	Dec 84	
4.2	Jun 85	
4.3	Aug 85	
4.4	Nov 85	
4.5	Jan 86	
4.6	May 86	
4.7	May 86	
4.8	Jan 87	[12]
4.9	May 87	
5.1	May 87	
5.2	Jun 87	
5.3	Oct 89	
5.4	Jun 90	
5.5	Jan 91	
5.6	Sep 91	[8]
5.7	Dec 93	[9]
6.1	Mar 97	[13]
6.2	Aug 01	[14]
6.3	Aug 03	[15]
6.4	Mar 06	[16]
7.0	May 00	[17]
8.1	Oct 07	[18]
8.2	Oct 14	[19]

break, a continuous hadron mass spectrum is obtained. At the time this was viewed as a limitation, but today it could be considered as the blueprint for a Lorentz covariant cluster fragmentation model. Unfortunately only few people were aware of that article in 1978.

By the suggestion of the head of the Lund group, Bengt E. Y. Svensson, two young PhD students, Bo Söderberg (who coined the JETSET name) and myself, were tasked with reproducing the work of Field and Feynman, and extending it to the analytical model developed in Lund. The practical conditions were not the best. There only existed one computer at Lund University with the capacity to run such programs, and it had a clock speed of approximately 1 MHz and a CPU memory of 1 MB. Worse, input was by a stack of punched cards, where each card corresponded to one 80-character line of Fortran 77 code. The card reader had a tendency to fail — at worst meaning a destroyed card — every few hundred cards, thereby favouring compact programming. Output was by line printer, some unpredictable 10 to 30 minutes later. The smallest error and you had to retype the affected card(s), reread the whole program and wait another 20 minutes. The Fortran language was new to us, but easy to learn. The CERN HBOOK histogramming package [27] was one of the few libraries available, and its user interface influenced how we thought about having a library of physics and service methods operating on a common event record.

It is maybe not surprising that the seniors, especially Bo Andersson, initially were quite hesitant whether there was any future in doing physics studies this way. Or that Bo Söderberg grew tired of it, leaving me to continue alone. After about a year conditions slowly started to improve, in particular with the introduction of simple (phone-line) terminals and the ability to edit and save programs on the computer.

While the fate of the nascent JETSET program still hang in the balance, the PETRA $\mathrm{e}^+\mathrm{e}^-$ collider at DESY started to produce results on jets, and notably found three-jet events [28], offering evidence for the existence of gluons. The experimental observations were backed up by comparisons with the event generator developed by Hoyer et al. [29] and, later, with the one by Ali et al. [30]. Both of these were based on the concept of independent fragmentation, wherein each of the q, $\overline{\mathrm{q}}$ and g jets are assumed to fragment symmetrically around a jet axis defined by the direction of the respective parton in the CM frame of the event. In Lund, instead another picture had been developed, string fragmentation. Here the confining colour field is approximated by a massless relativistic string, with gluons represented by pointlike momentum-carrying "kinks" [31]. For a $\mathrm{q}\overline{\mathrm{q}}\mathrm{g}$ event a colour field is then stretched from the q via the g and on to the $\overline{\mathrm{q}}$, with no string directly connecting the q and $\overline{\mathrm{q}}$. This leads to the latter interjet region being depleted of particle production, whereas the other two are enhanced.

A simple code to generate events according to this approach was written, and we studied how our model would affect event properties, notably the structure of the gluon jet [32]. Our interactions with the PETRA experimentalists led to the JADE collaboration being able to present first evidence for the string picture at the Moriond meeting in March 1980 [33], and this was followed by further studies supporting the string picture [34]. This breakthrough overcame the misgivings the seniors had had previously, and from then on event generator development came to play an increasing role in the Lund group activities. The string model was not generally accepted initially, however. One of the reasons was that the TASSO collaboration failed to reproduce the JADE results (unpublished, but widely known). It was only in 1982, when the CELLO collaboration noted that string fragmentation requires a bigger α_s than independent fragmentation to describe the same three-jet rate [35], that it became important for DESY to see the issue resolved. A new TASSO analysis effort then found a mistake in the previous one, that had masked effects in the data, and matters began to settle down [36].

Even before this happened, another development was that the newly-formed LEP collaborations mainly opted for Jetset in their QCD studies, rather than for the Hoyer et al. and Ali et al. programs. (And similarly at PEP and TRISTAN.) A reason was that the latter two programs had been written on the DESY mainframe, making extensive but inefficient use of existing DESY software, such that they did not fit in the smaller CPU memory of the CERN mainframe at the time. By contrast, Jetset was written from scratch by one person, in a more efficient and compact manner, and easily fitted. This experience came to set its mark on further Jetset/Pythia development, in that an ambition remained for programs to be designed to run standalone and have a modest footprint. Thereby the programs early on could be installed across the World, including places with modest computing support, and also rapidly could make its way onto personal computers when later these came along. Another relevant factor was that Lund was a remote isolated place in the days before Internet, making the effort spent on writing detailed manuals essential for successful usage elsewhere.

In Lund, there were now two lines of development. One was to make the string fragmentation model itself more sophisticated. This involved a number of topics, concerning flavour composition and in particular baryon production, the transverse broadening by nonperturbative and semiperturbative mechanisms, and the longitudinal sharing of momenta [37]. The Physics Reports written in 1982 [20] fairly well summarizes this work, and marks the end of the most intense period of string fragmentation development. But further studies have been made now and then over the years, some incorporated into the public code and others not. Of special practical importance is the extension to string topologies with arbitrarily many gluons between the quark and antiquark ends of a string [38], and the extension to Y-shaped "junction" topologies [39], where three strings meet, as can be the case if the three valence quarks of a proton are kicked out in different directions.

The other line was the extension of the e^+e^- machinery to other collision process types. The first of these came from a request/plea of the spokesperson of the European Muon Collaboration to provide a simulation of Deeply Inelastic Scattering. Thus the Lepto program [40] was born, in the first instance coded by me but soon taken over by co-student Gunnar Ingelman. It included NLO (next-to-leading order) corrections, i.e. combined the $\ell\mathrm{q} \to \ell\mathrm{q}$ processes with $\ell\mathrm{q} \to \ell\mathrm{qg}$ and $\ell\mathrm{g} \to \ell\mathrm{q}\overline{\mathrm{q}}$.

Lepto introduced a pattern that was to be repeated in the years to come: a process-specific code that uses (or not) perturbation theory to set up parton-level configurations with specified colour-flow string connections, followed by a call to the Jetset string fragmentation methods to handle the rest of the story. The Jetset code itself consisted of two parts, one being the dedicated setup of perturbative e^+e^- annihilation processes, the other the multipurpose fragmentation methods. (In my first contacts with Computer Physics Communications, in 1982, these two parts therefore were submitted separately, according to the author instructions of the time. Submission was by snail mail, and code was sent in on storage tapes.)

The next student to join was Hans-Uno Bengtsson, who began to study string effects in hadron collisions, i.e. how the colour connections between partons lead to some regions having an enhanced particle flow and others a depleted one. The program Compton considered topologies where a photon recoils against a quark or a gluon, whereas HIGH-$p_\perp$ addressed general 2 → 2 scatterings of partons. The latter in particular raised a completely new issue, namely that different colour flow topologies could interfere with each other, giving rise to a (positive or negative) fraction of the cross section that could not be associated with a well-defined colour topology. Hans-Uno used the different pole structures to find a sensible but not unique subdivision.

The multiplication of program names was a concern that Hans-Uno and I discussed and, both being avid readers of *The Histories* by Herodotus of Halicarnassus, we settled on the name PYTHIA for the combined framework for $\mathrm{pp}/\mathrm{p\overline{p}}$ collisions.

3 The formative years

In early 1983 Gunnar left for CERN and I for DESY, while Hans-Uno a year later moved to UCLA. During my time at DESY I was busy with extending string fragmentation to arbitrarily long colour chains, implementing second-order matrix elements (MEs) in $\mathrm{e^+e^-}$ annihilation, studying the relationship between hadronization models and α_s determinations, and supporting the early HERA experimental studies, wherein LEPTO came to be used, extended by Gunnar from QED to full electroweak boson exchange.

Then I moved to Fermilab just in time to go to the 1984 Snowmass meeting, where future SSC physics was intensely studied and discussed. The main two event generators were ISAJET [41] and FIELDAJET [42], where the former was publicly available and the latter not, but both with lots of results to show, and I learned from Frank Paige and Rick Field about them. Both programs were based on independent fragmentation, which in principle gave PYTHIA an edge. But this was moot, since PYTHIA lacked the parton showers that gave the other two programs realistic jet shapes. There were also other shortcomings that made PYTHIA unsuited for collisions at high energies.

By agreement with Hans-Uno, therefore I began to work towards making PYTHIA competitive. (Technology had progressed; in the autumn of 1984 it became possible to send e-mail and thereby code between Fermilab and UCLA, opening up for long-distance collaboration. Europe was still out of reach.) A first step was to include final-state parton showers, which was done by coding up two already existing algorithms, a “conventional” shower proposed by Kajantie and Pietarinen [43] and the coherent one of Marchesini and Webber [44]. Since they could also be used for $\mathrm{e^+e^-}$ studies, they were made part of JETSET.

The challenge, however, was initial-state showers. The few studies that had been made at that point, notably with COJETS [45], were based on forwards evolution, wherein the cascade was started at a low Q_0 scale and then traced towards larger Q scales. This means that the hard interaction is not predetermined, which in practice leads to low efficiencies. At Snowmass I had been asking parton distribution function (PDF) experts like Wu-Ki Tung about the prospects of backwards PDF evolution, from large to small Q scales, i.e. in some sense backwards in time, with the conclusion that no appropriate tools existed. The key point is that, even if the forwards-evolution splitting kernels are flavour-symmetric, the flavour content of the proton is not, and this must be reflected in the backwards evolution. My proposed solution was an algorithm where the PDF evolution equations are inverted to give a backwards evolution not only containing splitting kernels, like in forwards evolution, but also ratios of PDFs [46]. This approach still is a key component of PYTHIA, and is used in most other current generators.

One hope had been that the introduction of ISR and FSR should make it possible to describe data coming from the Sp$\overline{\text{p}}$S. This was not the case; notably the long tail towards large multiplicities in UA5 data [47]was absent in the generator, as were the strong UA5 long-distance forward–backward multiplicity correlations [48]. Frank Paige had explained the multi-Pomeron-based model in ISAJET, a purely soft approach also used elsewhere. On the other hand, there had also been interesting studies on Double Parton Scattering (DPS), wherein two parton–parton collisions occur in the same hadron–hadron one, but mainly viewed as a rare high-$p_{\perp}$ process [49, 50, 51, 52]. Finally, UA1 had found minijets down to $p_{\perp} = 5$ GeV [53], a practical rather than a physics cutoff. These considerations I brought together in a first model for MultiParton Interactions (MPIs) [54], wherein parton–parton collisions are assumed to be perturbatively calculable down to a tunable scale $p_{\perp\mathrm{min}}$, at the time 1.6 GeV. (To be changed somewhat over the years, e.g. as a function of the small-x behaviour of PDFs.) A variable number of MPIs is obtained by evolution downwards in $p_{\perp}$ from the maximal scale, and this variability is what is needed to describe the above-mentioned UA5 data. The single-interaction and DPS frameworks are recovered at large $p_{\perp}$ scales, while most MPIs occur at smaller $p_{\perp}$ values and then are not too dissimilar from the $p_{\perp} = 0$ soft Pomerons.

A key issue was how the colour flow between different MPIs is related, and this led to a picture where colour reconnection (CR) was allowed. Over the years to come, this MPI+CR model was to be made successively more sophisticated [22]. It may be the PYTHIA core component on which most effort has been spent. While initially met with considerable scepticism, today it is generally accepted and the approach has spread to other generators.

The third PYTHIA improvement, in addition to showers and MPIs, was that we added many new processes. Initially the program had only contained QCD and QED $2 \to 2$ processes, but now Z^0, $\mathrm{W}^{\pm}$ and H^0 production were added, singly, in pairs or together with a parton. A first few Beyond the Standard Model (BSM) processes were also added. All cross sections and differential decay distributions had to be coded by hand from formulae in the literature, sometimes with issues that needed to be sorted out with the respective authors, which made many new processes rather time-consuming to implement. In addition to the explicit hard processes, the MPI machinery allowed an inclusive description of inelastic nondiffractive events, to which elastic and very simple diffractive events could be added to obtain a description of all components of the total cross section.

By this evolution PYTHIA became competitive with the other generators on the market, and in some respects surpassed them. At Snowmass 1986 Hans-Uno and I could fully join and contribute to the SSC physics studies, such as Higgs searches. Getting to be used by the big experimental collaborations was a more gradual process. The most important step was the year-long 1990 “Aachen” workshop that marked the beginning in earnest of physics and detector studies for the LHC. Being at CERN in 1989–95, I got involved in just about all the different physics subgroups. Thus most of my time was spent to explain various physics aspects and to cater to the generator needs in the subgroups, e.g. by implementing new BSM processes, and that meant many new PYTHIA users. When the LHC detectors gradually were designed, PYTHIA thereby came to be the main generator used to study the performance under different assumptions, and this propagated on through the subsequent physics preparations and into the operations era.

LEP started running in 1989, so also preparations for that took some effort in the second half of the eighties. It was clear that parton showers would play a key role in order to produce multijet final states, but also that three-jet events would be a main staple of QCD studies. The Kajantie–Pietarinen algorithm did not contain coherence, and the Marchesini–Webber one did not cover the full three-jet phase space. Instead a somewhat simpler shower was developed [55]. It was a hybrid, involving evolution in m^2 but with angular-ordering cuts to ensure coherence, and as such with limitations. It had one redeeming feature for its day, however, in that it covered the full three-body phase space at a calculable rate that was slightly above the ME one, neglecting Sudakov factor [56] effects. The veto algorithm [16] could therefore be used to reduce the first emission rate down to the ME level, multiplied by a Sudakov that is fixed by the ME and the choice of evolution variable. The same "ME corrections" or "ME exponentiation" formalism has later been applied also to $p_\perp$-ordered showers. It has been extended to cover the MEs for essentially all SM and SUSY two-body decays that are followed by a gluon emission [57], which means that the default treatment of γ^*/Z^0, $\mathrm{W}^\pm$, t and H^0 decays is accurate to NLO. The same approach can also be used to exponentiate the MEs for single $\gamma^*/\mathrm{Z}^0/\mathrm{W}^\pm/\mathrm{H}^0$ production in association with a quark or gluon [58].

At the same time another shower was being developed, also in Lund (where I was 1985–89). The Leningrad group had found that the soft-gluon emission pattern around a $\mathrm{q\overline{q}g}$ topology could be viewed as a sum of radiation off two independent dipoles, qg and $\mathrm{g\overline{q}}$, mimicking the nonperturbative string picture [59]. Gösta Gustafson had realized that this offered a starting point to formulate a shower as a successive branching of dipoles [60], an idea that today is a standard choice for most shower algorithms, also the PYTHIA ones [61], with local variations. A student had been put to implement this approach, that was to become the ARIADNE program, but progress was slow. (Else this approach might have found its way into JETSET sooner.) It was only later, with a second student, Leif Lönnblad, that ARIADNE took off [62] and usually offered the best shower description of LEP data.

4 The convergence of PYTHIA and JETSET

In the late eighties Hans-Uno moved away from particle physics. With a sole developer/maintainer of both JETSET and PYTHIA, the two programs therefore could be increasingly coordinated. In addition the distinction between JETSET for $\mathrm{e^+e^-}$ and PYTHIA for $\mathrm{pp/p\overline{p}}$ began to crumble. For LEP 2 preparations it was necessary to implement $\mathrm{W^+W^-}$, $\gamma^*/\mathrm{Z}^0\,\gamma^*/\mathrm{Z}^0$, $\mathrm{H^0Z^0}$, etc. But these production processes were already available for $\mathrm{pp/p\overline{p}}$, and could trivially be extended to $\mathrm{e^+e^-}$. The same goes for the key LEP 1 process of Z^0 production and decay. Therefore PYTHIA was becoming the prime repository of (hard and soft) processes, plus ISR and MPI, with JETSET handling the subsequent hadronization, plus FSR. The old JETSET $\mathrm{e^+e^-}$ machinery lived on, since it did allow for arbitrary transverse and longitudinal $\mathrm{e^+e^-}$ beam polarization, and contained second-order matrix elements, but gradually faded away, and is not ported to PYTHIA 8.

Given the tighter integration, the first combined physics description and manual of both programs appeared in 1992, already then 280 pages long. It was gradually updated and extended over the years to come, reaching 480 pages (with same formatting; 580 in JHEP) in the final PYTHIA 6.4 article in 2006. Before then, versions of this evolving document only appeared as preprints, with hardcopies in steady demand at the CERN computing center. (Already from the mid-eighties the center had supplied ad hoc collections of separate pieces of documentation.) The size reflects not only the breadth of physics covered but also the access given to all methods and parameters. Furthermore it is important to offer alternatives in order to test models, such that one can establish not only what works but also what does not, and this adds to the size.

Finally, in 1996, the JETSET code was integrated into the PYTHIA package, and program elements renamed to adhere to PYTHIA conventions.

Integration of course was not the only theme, but also continued evolution and expansion in a number of respects. It would carry too far to give a full coverage of the evolution up to the end of PYTHIA 6, but some examples are given below, in no particular chronological order.

One of the prime objectives of LEP 2 was to determine the W mass. The fully hadronic decays $\mathrm{e}^+\mathrm{e}^- \to \mathrm{W}^+\mathrm{W}^- \to \mathrm{q}_1\overline{\mathrm{q}}_2\mathrm{q}_3\overline{\mathrm{q}}_4$ were expected to give about as small errors as the semileptonic $\mathrm{q}_1\overline{\mathrm{q}}_2\ell\nu_\ell$ ones. One study suggested that colour reconnection would completely mess up the hadronic channel [63], but Valery Khoze and I did a more detailed study, showing that the perturbative effects should be under control, whereas nonperturbative CR effects could give a mass uncertainty of order 40 MeV [64]. This was based on two new models wherein the space-time string overlap between the two W systems (including parton-shower effects) is traced, assuming similarities with flux lines either in type I or in type II superconductors.

Another limiting factor could be Bose-Einstein (BE) effects, also linking the two W systems. BE is not part of the standard simulation chain in Pythia. Like many other aspects of generator physics it is grounded in quantum mechanics, but it is a truly nonlocal phenomenon that cannot even to first approximation be reduced to a simple probabilistic step-by-step procedure, unlike parton showers or string fragmentation. One possible way out is to assign events a weight once the final hadron topology is known. In PYTHIA another approach has been implemented, in which the momenta of particles are shifted so as to change distributions from approximate phase space to the intended two-body correlation function. The shift of each particle is calculated as a vector sum of the shifts inside each pair of identical particles that the particle belongs to. Leif Lönnblad and I extended this approach to W pairs, trying a few alternatives, and came to the conclusion that W mass uncertainties from BE could be even somewhat larger than those from CR.

Both issues were studied at LEP 2. With predicted effects at the edge of detectability, it is maybe not surprising that different conclusions were reached. In the final combined analysis [65] there are convincing evidence for a CR rate in agreement with predictions, whereas no evidence was found for BE effects.

The photon is not as simple as it might seem at first glance, since it can fluctuate (electromagnetically) into a $\mathrm{q}\overline{\mathrm{q}}$ pair and then undergo strong interactions. A low-virtuality fluctuation has time to emit further gluons and become closely similar to vector mesons like the ρ^0, with which it shares quantum numbers, while high-virtuality ones remain of a perturbative character. The photon can also interact in its simple unresolved form, even if this is the lesser part of the total cross section at high energies. Together with Gerhard Schuler, a CERN fellow, a complete framework was developed for γp physics e.g. at HERA [66] and $\gamma\gamma$ one e.g. at LEP 2 [67]. The full framework contains several process types and interaction scales already for real photons, and when additionally the photons may be virtual the situation becomes even more complicated [68].

A very special Higgs production channel at LHC is WW or ZZ gauge boson fusion, since the colour singlet nature of the exchanged particles seemingly would imply that the central rapidity range of the event would be free of other particle production than that of the Higgs decay itself. This is clearly not the case when MPIs are considered, where further interactions are likely to involve colour exchange and thus span strings across the full rapidity range [69]. There is still a small probability of no MPIs, "the rapidity gap survival factor" [70]. Recently such MPI concepts were also used to describe the rate of jet production in diffractive events within the Pythia context [71].

Heavy flavours, from charm to top, involve their own sets of problems, that have been studied from different angles over the years. One interesting issue is the large charm vs. anticharm production asymmetries at fixed-target experiments, that can be understood in terms of the string topologies involved. In π^-p collisions with $\overline{\mathrm{u}}\mathrm{u} \to \overline{\mathrm{c}}\mathrm{c}$, for instance, the $\overline{\mathrm{c}}$ is colour-connected to the π^- beam remnant and pulled forwards by the string, while the c is connected to the p beam remnant and held back [72]. This gives rise to quite different momentum spectra, where some particle species thereby can take more momentum than the (anti)charm quark they come from. When strings are so short that they can collapse to a single particle, also the production rates become quite asymmetric. Asymmetries are smaller at higher energies, and for b quarks, but not such that they can always be neglected in CP violation studies, where they form a background.

Before the top was found, alternative scenarios had to be considered. A "light" top would be long-lived enough that top hadrons had time to form, while a heavy top would decay before that happens. With the latter scenario confirmed, the issue immediately arises that there is no unique set of colour singlet final-state particles that can be associated with the original colour triplet top quark. This leads to top mass uncertainties, e.g. from colour reconnection, that have been studied over the years [73, 74]. But it also leads to nontrivial angular distributions, e.g. in $\mathrm{e}^+\mathrm{e}^- \to \mathrm{t}\overline{\mathrm{t}} \to \mathrm{b}\mathrm{W}^+\overline{\mathrm{b}}\mathrm{W}^-$ the emission of gluons with energies above the top width is essentially uncorrelated between the b and $\overline{\mathrm{b}}$, while soft gluons and nonperturbative hadronization strongly correlate the two [75].

Programs for matrix-element calculations have existed since long, but often as private code in no shape to be run by anybody else. In the nineties this started to change. In the LEP 2 preparations, some 15 different codes were available for four-fermion final states, some tailor-made and others general purpose [76]. Since these did not address parton showers or hadronization, an ad hoc four-fermion interface was constructed to Pythia. But that still left open what to do with the explosion of processes to be studied at the LHC, and the increase of tools created to allow this study with improved precision. At the 2001 Les Houches meeting therefore a generic interface was developed between matrix element generators and shower+MPIs+hadronization programs [77]. This first Les Houches Accord consisted of two Fortran commonblocks. One with beam and generation strategy information. The other with event-by-event listing of the particles involved in the hard process, plus some weight and scale information. With the need to interoperate across languages, some years later the Les Houches Event Files offered a plain text alternative, initially carrying the same information [78] but gradually expanded to cater to increasing needs.

One of the BSM physics areas not implemented in Pythia for a long time was Supersymmetry (SUSY). It has been popular with theorists and experimentalists over the years, more so than the BSM models that actually were included. But it involves such an overwhelming set of parameters, particles, production processes and decay chains that realistic scenarios can become quite complex. It was also an area where most of the Isajet effort went in, so it was difficult to offer a competitive alternative. However, in 1994 Stephen Mrenna, then a Caltech postdoc, informed me that he was implementing SUSY within the Pythia framework. This SPythia code was initially presented on its own [79], but thereafter rapidly integrated into the regular distribution. Some years later Peter Skands, as master and PhD student, implemented lepton- and baryon-number-violating decays [39], thereby further extending the framework. He also took the initiative to the SUSY Les Houches Accord (SLHA) [80], that set a standard how couplings and particle properties could be transferred from spectrum calculators to event generators, and thereby greatly eased the task of setting up various SUSY scenarios.

5 The transition to C++

The Fortran language had dominate scientific computing in particle physics since early days, and for many the assumption was that this would continue, updated to more recent versions than the dominating Fortran 77 one. But in the nineties a growing group of forerunners were busily advocating more modern languages, notably C++. The campaign succeeded, and eventually the CERN management decided not only that C++ would become the main language for the LHC era, but also that the Fortran language would be phased out, to the extent that no Fortran compiler would be made available on CERN computers. This sent shock waves through the event generator community, needless to say, and we had to relate to a new reality in which we had been declared obsolete.

As it came to play out, the era of mainframe computing with proprietary expensive compilers was drawing to an end and, with the transition to farms running free Linux/GCC software, the Fortran language continued to be available. This did not change the fact that the next generation of event generator users in the experimental community would not be taught Fortran, and that it was believed to be easier for students with a C++ background to find jobs outside particle physics.

So, in early 1998, Leif Lönnblad and I began an intended project to convert at least most of the Pythia functionality to C++ within three years. We came from different backgrounds, with Leif being one of the early advocates of C++ and I having no previous experience. The idea, however, was that Leif would do most of the work, and we had ensured funding to this end. Unfortunately things did not work out as intended. Leif got involved in HERA studies and workshops, slowing progress. A postdoc, who was quite keen to get involved, used the opportunity more as a way to explore C++ than to deliver working code.

There were also differences of philosophy. Firstly, Leif wanted to construct a sophisticated and powerful framework, while I was asking for a simple and easily understood structure, and we never managed to find a common middle ground. Secondly, after some years Leif made an agreement with the Herwig people that the then existing Pythia 7 base code would be renamed into ThePEG and form a common generator-neutral platform for both Herwig++ [81] and Pythia plugin modules, and Sherpa [82] ones if that group joined. The advantage would be that a user would only need to learn one platform, and

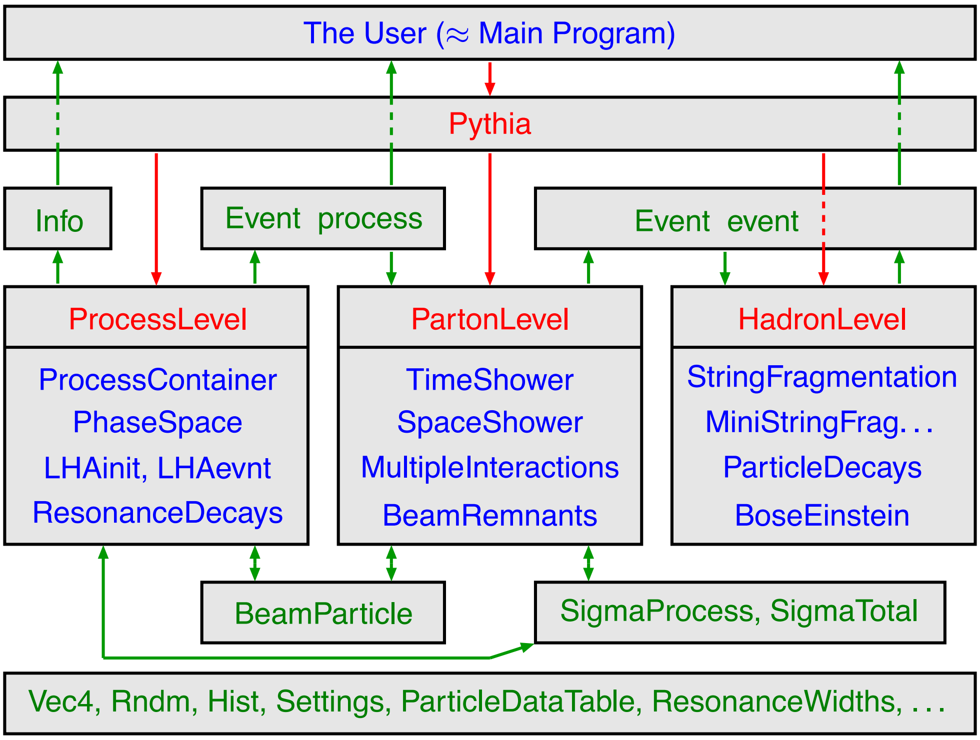

Figure 1: Schematic view of the code structure in PYTHIA 8 as of 2007.

that it would become simple to mix different physics models. This ran counter to my sentiments, that there is an intrinsic value in having completely independent codes to compare, to reduce the risks of common bugs, and that maybe not all combinations of models should be encouraged.

With progress stalled, the future looked bleak. Opportunity came in an unexpected way, namely that Lund University underwent an economical crisis, which also strongly affected our group. It therefore became feasible, and even encouraged, for me to go away for a longer period. The SFT group at CERN, in charge of developing common scientific software such as ROOT [83] and GEANT [84], offered to take me in for three years of code writing. Part of the deal was that this would not be just a salvaging operation of PYTHIA 7 but a fresh new start.

Thus, 2004–08 (with an unintended break of a few months) I worked on the new PYTHIA 8 code close to full-time. It was a fortunate timing, in between LEP and LHC, meaning less "distraction" from physics questions and workshops than normally. Meanwhile Stephen and Peter largely took care of PYTHIA 6 and Tevatron support.

The design philosophy was to keep the basic code as simple as possible. It was new code written from scratch, with few exceptions, but clearly inspired by strengths and shortcomings of the existing Fortran code.

A schematic view of the relationship between the new classes is shown in Fig. 1. Some generic facilities, like four-vectors and the settings and particle data databases, can be used just about anywhere. Else most classes occur in a hierarchy, with `Pythia` on top, and below that three class sets representing main stages in the evolution of an event, from hard perturbatively calculable processes to soft nonperturbative ones.

- The `ProcessLevel` administrates the choice of a hard process by a combination of matrix element expressions and phase space selection. Most of the processes in PYTHIA 6 were taken over. Input of externally generated events via the Les Houches accord, an afterthought in PYTHIA 6, was here a standard option from the beginning. The operation of internal processes was even adjusted to conform. Specifically, whereas previously weak decay chains like $\mathrm{t} \to \mathrm{bW}^+ \to \mathrm{bq\overline{q}}$ were only generated after showers in previous steps had run their course, now they are generated already at the process level. The decays are stored as part of the `process` hard-process event record, from which information is then fetched in as needed later. For the few but important "soft" processes, such as inelastic nondiffractive events, the process-level stage is rather minimal, with the real action beginning later.

- The `PartonLevel` traces the continued evolution down to lower scales and higher partonic multiplicities, by including the effects of parton showers on the hard process, by adding MPIs (in the figure called multiple interactions), and eventually by adding beam remnants. Administratively, ISR, FSR and MPI are all three generated in an interleaved manner, in order of decreasing $p_\perp$ scales. The philosophy is that the harder activity sets the boundary conditions for the softer one, irrespective of naive time ordering. Apart from the hard events discussed so far, also soft events can here be imparted with MPI activity, which may give ISR/FSR in its turn. Other physics mechanisms, such as colour reconnection, can also occur at this stage.

- The final `HadronLevel` step turns partons into hadrons by string fragmentation, and takes care of subsequent decays of hadrons and leptons. This step can also include other nonperturbative physics, such as models for Bose-Einstein correlations.

The main output of the generation is the `event` event record, that documents both initial, intermediate and final particles generated in the course of the three stages above.

Many generators are intended to be run by a set of input cards to a single executable that, once built, does not need to be touched again. PYTHIA 8 can be run in this way, but it is not the only one. Rather, it is the ability to use the program in different contexts that lends real power. A set of example main programs is provided to illustrate this flexibility. The main program can contain quite sophisticated event analyses, making use of the power of having the full event history available, in combination with various analysis codes, notably a custom interface to `FastJet` for jet clustering [85]. In addition to a multitude of settings that can affect the program execution, there is also the ability to insert external code at critical points along the generation chain, by plugins or user hooks. The former offer a way to replace some of the standard tasks by your own code, notably for parton showers or random numbers. The user hooks calls are interspersed with the normal code and allow optional extensions of it. Typically a hook is intended to allow for simple decisions, e.g. to veto some parton-shower emissions. As it happens, there has been a steady demand for more user hooks.

The three-year period was sufficient to have PYTHIA 8.100 up and running, with about as much LHC physics functionality as the PYTHIA 6 code had. But time did not permit to convert everything, with γp, $\gamma\gamma$, SUSY and Technicolor physics being among the afflicted. And some obsolete options were scrapped altogether.

6 Current activities

The Pythia 8 code has continued to expand after the original release, and today more persons are involved in development work in and around Pythia than ever before. Some main themes are described in the rest of this section.

As perturbative calculational capabilities have progressed, it has become possible to generate processes with more final-state particles, not only to leading order (LO) but also routinely to NLO, and for some processes to even higher orders. Nevertheless, it is rarely if ever possible to use a pure ME approach to describe all perturbative activity down to the hadronization scale, as required to provide a consistent description, so one must still combine MEs with parton showers. This field of activity is today called Match and Merge (M&M), where match refers to providing a smooth transition from fixed-multiplicity MEs above some scale to showers below that, and merge to the consistent combination of different ME multiplicities. It can be argued that this has been the main theme of event-generator development work in the last twenty years. With some forerunners, like the already-mentioned exponentiation of matrix elements for the first emission, the real beginning was a 1999/2000 LEP 2 workshop. Out of discussions there sprung the idea that LO MEs of different multiplicities could be combined consistently, provided that they were corrected with Sudakov form factors describing the no-emission virtual corrections. Two related approaches were developed, the CKKW one [86], where the Sudakovs are obtained from the analytical expression, and the CKKW-L one [87], where explicit "trial showers" are used to generate the Sudakovs. Today the latter approach has become the norm, but has branched out in several variants.

M&M methods were not part of the distributed Pythia 6 code, but a number of studies were still done using different add-on codes [88, 89], e.g. as part of MadGraph [90]. In Pythia 8 several such methods are part of the distribution, but usually as plugins rather than in the core library. However, in 2011 Leif Lönnblad together with then-student Stefan Prestel began to develop and implement M&M methods in the Pythia 8 core, addressing detailed issues such as how interleaving of showers with MPIs should be taken into account [91]. (Which involved the creation of a whole new `PartonLevel` instance to run trial showers and MPIs.) The development extended to include also usage of NLO MEs [92, 93] in a few different approaches. This code is still actively developed to cover more kinds of processes and higher multiplicities, among others. Combined with the ones previously mentioned, there is currently approximately ten different M&M schemes available to play with. In addition, M&M in external ME programs like MadGraph and the PowHeg Box [94] are also commonly used.

With the ME part of the story increasingly well modelled, it is natural also to develop the shower part to have higher precision. Early on Pythia 8 was opened up to allow external shower programs to be linked in as a replacement for the native ones, and over the years this has been used for some studies. Of special interest are the Vincia [95] and Dire [96] projects. Both have the ambition to raise showers to next-to-leading or even next-to-next-to leading logarithmic precision. Vincia, by Peter Skands and coworkers, is the older of the two. Over the years it has been used to test out a number of new approaches, such as smooth ordering, sector showers, antenna showers (for FSR+ISR), helicity-dependent showers, and iterated ME corrections. Dire is a rather unusual project, in that the same shower algorithm has been coded twice, once for Sherpa by Stefan Höche and once for

Pythia by Stefan Prestel, so as to reduce the risk of bugs. Unlike most other codes it gives variable and even negative weights already at leading log.

Both Vincia and Dire are large programs in their own right, not only by the core evolution code itself, but by the environment of matrix elements, splitting libraries, M&M machineries and more. The advantage of having been freestanding is that development and new releases has been decoupled from the Pythia ones. The disadvantage is that the threshold for users to try out these showers is higher. For the upcoming Pythia 8.3 release, therefore, they will become part of this distribution. They will still continue to be vigorously developed as identifiable subpackages.

Also the default Pythia parton showers have continued to evolve over the years, e.g. by the optional emission of weak gauge bosons [97] and by a framework for showers in various Hidden Valley scenarios [98]. The latter also includes hadronization in the secluded sector, and decay back into the visible sector. Similarly, production of long-lived coloured particles, such as squarks and gluinos in some scenarios, combine showers and hadronization. First these particles can radiate, then hadronize into so-called R-hadrons that propagate some distance, then decay in a process that again will involve showers and a new hadronization step.

Traditionally the emphasis of Pythia has been on incoming hadron or lepton beams, including somewhat special cases like photons and Pomerons. But already in the mid-eighties a model Fritiof [99] and related program [100] was developed for hadronic reactions, with generalization to heavy-ion collisions. In it nucleons could get an increasingly excited mass by successive collisions. The resulting states would undergo Jetset string fragmentation after the two nuclei had passed through each other. The model was quite successful for fixed-target energies, but perturbative parton–parton scatterings were difficult to include, and therefore it could not reliably be used at higher energies. A few years ago heavy-ion activities were started up again, with the Angantyr model [101]. It is based on the conventional MPI model for pp collisions, and allows for a nucleon to undergo successive collisions, where one is of the conventional pp type and the rest of them can be viewed as a Pomeron taken out of the nucleon colliding with a nucleon from the other nucleus. This is similar to how the beam-remnant machinery handles MPIs in ordinary pp collisions, such that not all strings stretch out all the way to the edge of the allowed rapidity range but stay more centrally. The model also includes shove, that two strings can repel each other and thereby give collective flow and azimuthal anisotropies, and ropes, that two strings can overlap to give a higher string tension that favours the production of more strangeness. The objective is to see how far it is possible to go with the description of heavy-ion collisions without invoking the existence of a quark–gluon plasma. Thus the development of Angantyr and related aspects will be a central undertaking for the coming years.

A more recent project has been to pin down the space–time structure of the hadronization process in pp collisions [102]. This is a first step towards modelling hadronic rescattering, initially for pp but later possibly also for heavy-ion collisions. An important component is the modelling of hadronic collisions from threshold energies upwards, which is not possible currently, and where the collision energy and hadron types can be changed flexibly. Such a machinery could also find other applications, e.g. for cosmic-ray cascades in the atmosphere.

Other physics studies are ongoing within the Pythia context, like photoproduction and $\gamma\gamma$ physics, diffraction and total cross sections, neutrino interactions, Dark Matter processes and other BSM physics, coalescence processes for deuterium production, production of charmonium and bottomonium in showers, new possibilities for DPS studies, and more. Projects like that will always show up and hopefully leave their imprint in the code available for users.

Most of the PYTHIA model components involve free parameters that have to be determined by comparisons with data. The total number of parameters is quite large and typically these are correlated in nontrivial ways. Therefore, from the early days onwards, the production of internally consistent tunes has been a recurring activity. This is a task performed both by PYTHIA members and by the various experimental collaborations. Unfortunately the latter kind of tunes tend to be restricted to data from the own experiment, and sometimes only to a subset, e.g. either to minimum-bias or to high-$p_{\perp}$ jet data. That way agreement can be improved for the tasks at hand, at the expense of worse agreement elsewhere. Global detector-agnostic tunes therefore have a relevant role to play. Current default values are based on the Monash 2013 global tune [103], which also has served as a starting point for other tunes.

In addition to physics modelling and studies, there is also a significant upgrade in the works, namely that the upcoming PYTHIA 8.3 release will be based on C++11 rather than on C++98. This means e.g. that smart pointers, container loops and function wrappers will come into use, initially at a few places in the existing code but likely to be more common in new code. Some of the other new C++11 features we have decided not to use, so as not to introduce unnecessary complication and make code less transparent.

There are also other changes to the code, some of which are unrelated to the language upgrade, but suitable to implement when backwards compatibility is anyway (mildly) broken. One such is a new `InfoHub` class that includes several other service classes, thereby reducing the number of arguments needed to pass around (pointers to) other classes. This is combined with a new `PhysicsBase` base class, from which several of the physics classes are derived, with automatic import of `InfoHub`, and an option to set up methods that are called before and after each event has been generated. The intention is also to improve the parton-shower interface, to streamline the VINCIA and DIRE integration. There will be a new interface for various string interaction models.

The XML/HTML documentation has expanded significantly since the original 8.1 release, and the separate pages are now being reordered for better cohesion. A searchable index of the example main programs is added. In the future the pages should gradually be expanded with more descriptions of the physics involved, eventually leading up to a new publishable long 8.3 manual. The PHP version is discontinued, since it would require double work to maintain within the intended new structure.

7 Summary and outlook

As we have seen, the JETSET/PYTHIA project has expanded, from a humble project to study some simple distributions within a single jet, to a code with intentions to be relevant for essentially all areas of high-energy particle physics.

As a natural consequence, the PYTHIA code size has been steadily increasing over the years; obviously with PYTHIA 8 starting over from scratch but then rapidly overtaking PYTHIA 6. The discovery of the string effect was based on a code with a total size of 1 000

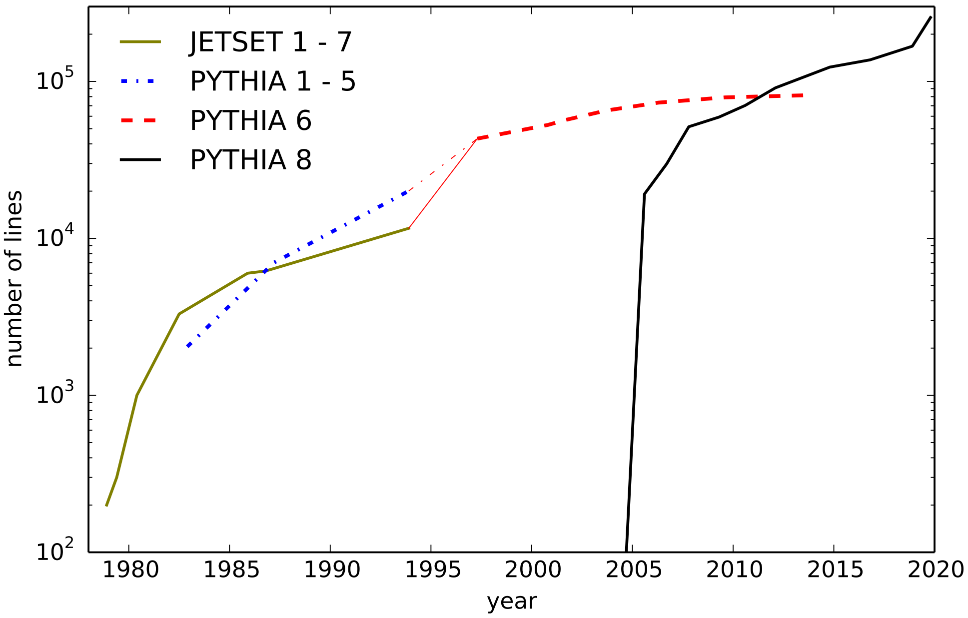

Figure 2: Number of lines in the program codes as a function of time. Snapshots in time are connected by straight lines. Thin lines around 1995 mark the merger of Pythia and Jetset.

lines, while projections for the 8.3 release hover around 250 000 ones. A big jump from the current 8.2 size is explained by the inclusion of the Vincia and Dire shower codes. This evolution is shown in Fig. 2. It is largely based on what the `wc` command gives, which include both comment and blank lines, and is only for the core program, which for the C++ versions are the `include/Pythia8` and `src` subdirectories. The actual Pythia 8 code distribution is larger, with example programs, parton distribution function data files, manual pages, and more. So far the original organizational structure of the code has been possible to extend gradually and reasonably smoothly, but this may not always be the case.

The current and previous versions of the Pythia code, along with auxiliary documentation and files, and relevant presentations by team members, can be found at the Pythia webpage

`http://home.thep.lu.se/Pythia`

In one respect Pythia 8 is still trailing Pythia 6, namely in the size of the manual. The current XML/HTML-based manual does document all settings and all user-accessible methods, but is rather brief in the physics descriptions. Most is documented in separate physics articles, of course, but from these it is not always possible to get a coherent view over which different ideas have been implemented and combined how. One of the projects for the future is to improve the physics documentation, both on the separate HTML web pages and as a combined overview.

The size of the development team has fluctuated over the years, but is currently rising at a steady pace. It remains quite Lund-centric, with most members being either former (Christian Bierlich, Leif Lönnblad, Stefan Prestel, Peter Skands, myself) or current (Christine Rasmussen) PhD students. Others have been recruited as postdocs (Ilkka Helenius) or short-term students (Nishita Desai, Philip Ilten) within the MCnet collaboration of event generator authors. Only one person (Stephen Mrenna) came in directly from the outside. Note that the number of members does not translate directly into man-years of PYTHIA development, since for many this is one activity among others.

In the LEP and early LHC days I was alone to answer all questions people might have, which took quite a significant chunk out of my daily work. The bigger size is quite helpful in this respect, both for sharing the basic questions and for assigning the more specialized ones according to our respective areas of expertise. By chance there is also a good spread of LHC contacts, in that Stephen is a CMS member and Phil an LHCb one, while Stefan is an ATLAS "Analysis Consultant & Expert" and Christian is closely working together with ALICE people.

The organizational structure has not quite kept pace with the expanding team. By tradition things are run informally, and people mainly contribute to their areas of interest (or those of their supervisor). The ambition of monthly Skype/Vidyo meetings has been limping. A 2019 week-long meeting at Monash University (Melbourne, Australia) was the first-ever organized in-person chance to discuss the near- and far-term development of the program. The former is reflected in the strategy for a transition from C++98 to C++11, now well along the way, and the subsequent release of PYTHIA 8.3. The latter involve plans for the next manual but also less conclusive discussions on the future organizational structure, including issues such as having internal reviewers before new code is added to the public version. The key message, however, is that the PYTHIA collaboration intends to be in there for the long haul.

We have already touched on areas of ongoing activities, such as heavy-ion physics, improved parton showers, or rescattering. More areas are likely to arise in the future, unknown now exactly which, but partly related to future directions in experimental physics. FCC-hh is a natural evolution from LHC and FCC-ee from LEP, so can already be run with reasonable expectations of reliably foretelling what to expect. For CLIC the challenge is the large background from beamstrahlung interactions, which partly can be modelled already now but may require further development. More work would definitely be needed for the EIC and FCC-eh, both to consistently model the transition region between photoproduction and DIS and to handle nuclear effects.

Finally, it is important to note that PYTHIA has played a dual role throughout its history. On the one hand, we have striven for increased precision/accuracy in predicting/describing experimental data, thereby offering an indispensable tool for the experimental community. But, on the other hand, it has also been a way to explore new ideas, say in topics that may be beyond perturbative control, and in the end these have often been as important for the advancement of the field. The hope and aim is that this dual nature will carry PYTHIA on into the future.

Acknowledgements

The progress outlined in this article would not have been possible without the collaboration of a large number of persons, including but not limited to past and present PYTHIA collaboration members, who have contributed both ideas and code. Apologies that not all such contributions are documented here.

Work supported in part by the Swedish Research Council, contract number 2016-05996, and in part by the MCnetITN3 H2020 Marie Curie Innovative Training Network, grant agreement 722104. This project has also received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, grant agreement No 668679.

References

[1] T. Sjöstrand, B. Söderberg, A Monte Carlo program for quark jet generation, LU TP 78-18 (1978).

[2] T. Sjöstrand, A Monte Carlo program for heavy quark jet generation, LU TP 79-8 (1979).

[3] T. Sjöstrand, A Monte Carlo Program for Quark and Gluon Jet Generation, LU TP 80-3 (1980).

[4] T. Sjöstrand, The Lund Monte Carlo for Jet Fragmentation, Comput. Phys. Commun. 27 (1982) 243.

[5] T. Sjöstrand, The Lund Monte Carlo for e+ e- Jet Physics, Comput. Phys. Commun. 28 (1983) 229.

[6] T. Sjöstrand, The Lund Monte Carlo for Jet Fragmentation and e+ e- Physics: Jetset Version 6.2, Comput. Phys. Commun. 39 (1986) 347–407.

[7] T. Sjöstrand, M. Bengtsson, The Lund Monte Carlo for Jet Fragmentation and e+ e- Physics. Jetset Version 6.3: An Update, Comput. Phys. Commun. 43 (1987) 367.

[8] T. Sjöstrand, PYTHIA 5.6 and JETSET 7.3: Physics and manual, CERN-TH-6488-92 (1992).

[9] T. Sjöstrand, High-energy physics event generation with PYTHIA 5.7 and JETSET 7.4, Comput. Phys. Commun. 82 (1994) 74–90.

[10] H. U. Bengtsson, The Lund Monte Carlo for High p_T Physics, Comput. Phys. Commun. 31 (1984) 323.

[11] H. U. Bengtsson, G. Ingelman, The Lund Monte Carlo for High p_T Physics, Comput. Phys. Commun. 34 (1985) 251.

[12] H.-U. Bengtsson, T. Sjöstrand, The Lund Monte Carlo for Hadronic Processes: Pythia Version 4.8, Comput. Phys. Commun. 46 (1987) 43.

[13] T. Sjöstrand, P. Edén, C. Friberg, L. Lönnblad, G. Miu, S. Mrenna, E. Norrbin, High-energy physics event generation with PYTHIA 6.1, Comput. Phys. Commun. 135 (2001) 238–259.

[14] T. Sjöstrand, L. Lönnblad, S. Mrenna, PYTHIA 6.2: Physics and manual, http://arxiv.org/abs/hep-ph/0108264.

[15] T. Sjöstrand, L. Lönnblad, S. Mrenna, P. Z. Skands, Pythia 6.3 physics and manual, http://arxiv.org/abs/hep-ph/0308153.

[16] T. Sjöstrand, S. Mrenna, P. Z. Skands, PYTHIA 6.4 Physics and Manual, JHEP 05 (2006) 026.

[17] M. Bertini, L. Lönnblad, T. Sjöstrand, PYTHIA version 7-0.0: A Proof of concept version, Comput. Phys. Commun. 134 (2001) 365–391.

[18] T. Sjöstrand, S. Mrenna, P. Z. Skands, A Brief Introduction to PYTHIA 8.1, Comput. Phys. Commun. 178 (2008) 852–867.

[19] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, P. Z. Skands, An Introduction to PYTHIA 8.2, Comput. Phys. Commun. 191 (2015) 159–177.

[20] B. Andersson, G. Gustafson, G. Ingelman, T. Sjöstrand, Parton Fragmentation and String Dynamics, Phys. Rept. 97 (1983) 31–145.

[21] A. Buckley, et al., General-purpose event generators for LHC physics, Phys. Rept. 504 (2011) 145–233.

[22] T. Sjöstrand, The Development of MPI Modeling in Pythia, Adv. Ser. Direct. High Energy Phys. 29 (2018) 191–225, http://arxiv.org/abs/1706.02166.

[23] B. Andersson, G. Gustafson, C. Peterson, A Quark Parton Model for Hadronic Fragmentation Distributions, Phys. Lett. 71B (1977) 337–341.

[24] B. Andersson, G. Gustafson, C. Peterson, A Semiclassical Model for Quark Jet Fragmentation, Z. Phys. C1 (1979) 105.

[25] R. D. Field, R. P. Feynman, A Parametrization of the Properties of Quark Jets, Nucl. Phys. B136 (1978) 1, [,763(1977)].

[26] X. Artru, G. Mennessier, String model and multiproduction, Nucl. Phys. B70 (1974) 93–115.

[27] R. Brun, I. Ivanchenko, P. Palazzi, HBOOK: Histoprogramming, Fitting and Data Presentation Package: Users Guide, Version 3.

[28] R. Brandelik, et al., Evidence for Planar Events in e+ e- Annihilation at High-Energies, Phys. Lett. 86B (1979) 243–249.

[29] P. Hoyer, P. Osland, H. G. Sander, T. F. Walsh, P. M. Zerwas, Quantum Chromodynamics and Jets in e+ e-, Nucl. Phys. B161 (1979) 349–372.

[30] A. Ali, J. G. Korner, Z. Kunszt, E. Pietarinen, G. Kramer, G. Schierholz, J. Willrodt, QCD Predictions for Four Jet Final States in e+ e- Annihilation, Nucl. Phys. B167 (1980) 454–478.

[31] B. Andersson, G. Gustafson, Semiclassical Models for Gluon Jets and Leptoproduction Based on the Massless Relativistic String, Z. Phys. C3 (1980) 223.

[32] B. Andersson, G. Gustafson, T. Sjöstrand, How to Find the Gluon Jets in e+ e- Annihilation, Phys. Lett. B94 (1980) 211–215.

[33] W. Bartel, A. Petersen, Results From Jade, in: Elementary Constituents and Hadronic sStructure. Proceedings, 15th Rencontres de Moriond, Les Arcs, France, March 9-21, 1980. Vol. 1, 1980, pp. 505–520.

[34] W. Bartel, et al., Experimental Study of Jets in electron - Positron Annihilation, Phys. Lett. B101 (1981) 129–134.

[35] H. J. Behrend, et al., The Influence of Fragmentation Models on the Determination of the Strong Coupling Constant in e^+e^- Annihilation Into Hadrons, Nucl. Phys. B218 (1983) 269–288.

[36] M. Althoff, et al., A Study of Three Jet Events in e+ e- Annihilation Into Hadrons at 34.6-GeV Center-Of-Mass Energy, Z. Phys. C29 (1985) 29.

[37] B. Andersson, G. Gustafson, B. Soderberg, A General Model for Jet Fragmentation, Z. Phys. C20 (1983) 317.

[38] T. Sjöstrand, Jet Fragmentation of Nearby Partons, Nucl. Phys. B248 (1984) 469–502.

[39] T. Sjöstrand, P. Z. Skands, Baryon number violation and string topologies, Nucl. Phys. B659 (2003) 243.

[40] G. Ingelman, T. Sjöstrand, A Monte Carlo Program for Leptoproduction, LU TP 80-12 (1980).

[41] F. E. Paige, S. D. Protopopescu, ISAJET 5.02: A Monte Carlo Event Generator for p p and anti-p p Interactions, Conf. Proc. C850809 (1985) 41.

[42] R. D. Field, Jets and Event Topologies at the SSC, in: Design and Utilization of the Superconducting Super Collider. Proceedings, 1984 Summer Study, Snowmass, USA, June 23 - July 13, 1984, 1984.

[43] K. Kajantie, E. Pietarinen, Improved Evolution Equations for QCD, Phys. Lett. 93B (1980) 269–273.

[44] G. Marchesini, B. R. Webber, Simulation of QCD Jets Including Soft Gluon Interference, Nucl. Phys. B238 (1984) 1–29.

[45] R. Odorico, Cojets: A Monte Carlo Program Simulating QCD in Hadronic Production of Jets and Heavy Flavors with Inclusion of Initial QCD Bremsstrahlung, Comput. Phys. Commun. 32 (1984) 139.

[46] T. Sjöstrand, A Model for Initial State Parton Showers, Phys. Lett. B157 (1985) 321–325.

[47] G. J. Alner, et al., Scaling Violation Favoring High Multiplicity Events at 540-GeV CMS Energy, Phys. Lett. B138 (1984) 304–310.

[48] R. E. Ansorge, et al., Charged Particle Correlations in $\bar{P}P$ Collisions at c.m. Energies of 200-GeV, 546-GeV and 900-GeV, Z. Phys. C37 (1988) 191–213.

[49] P. V. Landshoff, J. C. Polkinghorne, Calorimeter Triggers for Hard Collisions, Phys. Rev. D18 (1978) 3344.

[50] C. Goebel, F. Halzen, D. M. Scott, Double Drell-Yan Annihilations in Hadron Collisions: Novel Tests of the Constituent Picture, Phys. Rev. D22 (1980) 2789.

[51] N. Paver, D. Treleani, Multi - Quark Scattering and Large p_T Jet Production in Hadronic Collisions, Nuovo Cim. A70 (1982) 215.

[52] B. Humpert, Are there multi - quark interactions?, Phys. Lett. B131 (1983) 461–467.

[53] F. Ceradini, Study of minimum bias trigger events at s**(1/2) = 0.2-TeV to 0.9-TeV with magnetic and calorimetric analysis at the CERN proton - anti-proton collider, Conf. Proc. C850718 (1985) 363.

[54] T. Sjöstrand, Multiple Parton-Parton Interactions in Hadronic Events, Oregon Workshop on Super High Energy Physics, FERMILAB-PUB-85-119-T (1985).

[55] M. Bengtsson, T. Sjöstrand, Coherent Parton Showers Versus Matrix Elements: Implications of PETRA - PEP Data, Phys. Lett. B185 (1987) 435.

[56] V. V. Sudakov, Vertex parts at very high-energies in quantum electrodynamics, Sov. Phys. JETP 3 (1956) 65–71, [Zh. Eksp. Teor. Fiz.30,87(1956)].

[57] E. Norrbin, T. Sjöstrand, QCD radiation off heavy particles, Nucl. Phys. B603 (2001) 297–342.

[58] G. Miu, T. Sjöstrand, W production in an improved parton shower approach, Phys. Lett. B449 (1999) 313–320.

[59] Y. I. Azimov, Y. L. Dokshitzer, V. A. Khoze, S. I. Troian, The String Effect and QCD Coherence, Phys. Lett. 165B (1985) 147–150.

[60] G. Gustafson, Dual Description of a Confined Color Field, Phys. Lett. B175 (1986) 453.

[61] T. Sjöstrand, P. Z. Skands, Transverse-momentum-ordered showers and interleaved multiple interactions, Eur. Phys. J. C39 (2005) 129–154.

[62] L. Lönnblad, ARIADNE version 4: A Program for simulation of QCD cascades implementing the color dipole model, Comput. Phys. Commun. 71 (1992) 15–31.

[63] G. Gustafson, U. Pettersson, P. M. Zerwas, Jet Final States in W W Pair Production and Color Screening in the QCD Vacuum, Phys. Lett. B209 (1988) 90–94.

[64] T. Sjöstrand, V. A. Khoze, On Color rearrangement in hadronic W^+W^- events, Z. Phys. C62 (1994) 281–310.

[65] S. Schael, et al., Electroweak Measurements in Electron-Positron Collisions at W-Boson-Pair Energies at LEP, Phys.Rept. 532 (2013) 119–244.

[66] G. A. Schuler, T. Sjöstrand, Towards a complete description of high-energy photoproduction, Nucl. Phys. B407 (1993) 539–605.

[67] G. A. Schuler, T. Sjöstrand, A Scenario for high-energy gamma gamma interactions, Z. Phys. C73 (1997) 677–688.

[68] C. Friberg, T. Sjöstrand, Total cross-sections and event properties from real to virtual photons, JHEP 09 (2000) 010.

[69] Y. L. Dokshitzer, V. A. Khoze, T. Sjöstrand, Rapidity gaps in Higgs production, Phys. Lett. B274 (1992) 116–121.

[70] J. D. Bjorken, Rapidity gaps and jets as a new physics signature in very high-energy hadron hadron collisions, Phys. Rev. D47 (1993) 101–113.

[71] C. O. Rasmussen, T. Sjöstrand, Hard Diffraction with Dynamic Gap Survival, JHEP 02 (2016) 142.

[72] E. Norrbin, T. Sjöstrand, Production and hadronization of heavy quarks, Eur. Phys. J. C17 (2000) 137–161.

[73] P. Skands, D. Wicke, Non-perturbative QCD effects and the top mass at the Tevatron, Eur. Phys. J. C52 (2007) 133–140.

[74] S. Argyropoulos, T. Sjöstrand, Effects of color reconnection on $t\bar{t}$ final states at the LHC, JHEP 1411 (2014) 043.

[75] V. A. Khoze, T. Sjöstrand, Color correlations and multiplicities in top events, Phys. Lett. B328 (1994) 466–476.

[76] D. Yu. Bardin, et al., Event generators for WW physics, in: 3rd CERN Workshop on LEP2 Physics Geneva, Switzerland, November 2-3, 1995, 1997, pp. 3–101, [,3(1997)], http://arxiv.org/abs/hep-ph/9709270.

[77] E. Boos, et al., Generic User Process Interface for Event Generators, in: Physics at TeV colliders. Proceedings, Euro Summer School, Les Houches, France, May 21-June 1, 2001, 2001, http://arxiv.org/abs/hep-ph/0109068.

[78] J. Alwall, et al., A Standard format for Les Houches event files, Comput. Phys. Commun. 176 (2007) 300–304.

[79] S. Mrenna, SPYTHIA, a supersymmetric extension of PYTHIA 5.7, Comput. Phys. Commun. 101 (1997) 232–240.

[80] P. Z. Skands, et al., SUSY Les Houches accord: Interfacing SUSY spectrum calculators, decay packages, and event generators, JHEP 07 (2004) 036.

[81] M. Bähr, et al., Herwig++ Physics and Manual, Eur. Phys. J. C58 (2008) 639–707.

[82] T. Gleisberg, S. Hoeche, F. Krauss, A. Schalicke, S. Schumann, J.-C. Winter, SHERPA 1. alpha: A Proof of concept version, JHEP 02 (2004) 056.

[83] I. Antcheva, et al., ROOT: A C++ framework for petabyte data storage, statistical analysis and visualization, Comput. Phys. Commun. 180 (2009) 2499–2512.

[84] S. Agostinelli, et al., GEANT4: A Simulation toolkit, Nucl. Instrum. Meth. A506 (2003) 250–303.

[85] M. Cacciari, G. P. Salam, G. Soyez, FastJet User Manual, Eur. Phys. J. C72 (2012) 1896.

[86] S. Catani, F. Krauss, R. Kuhn, B. R. Webber, QCD matrix elements + parton showers, JHEP 11 (2001) 063.

[87] L. Lönnblad, Correcting the colour-dipole cascade model with fixed order matrix elements, JHEP 05 (2002) 046.

[88] S. Mrenna, P. Richardson, Matching matrix elements and parton showers with HERWIG and PYTHIA, JHEP 05 (2004) 040.

[89] J. Alwall, et al., Comparative study of various algorithms for the merging of parton showers and matrix elements in hadronic collisions, Eur. Phys. J. C53 (2008) 473–500.

[90] J. Alwall, P. Demin, S. de Visscher, R. Frederix, M. Herquet, F. Maltoni, T. Plehn, D. L. Rainwater, T. Stelzer, MadGraph/MadEvent v4: The New Web Generation, JHEP 09 (2007) 028.

[91] L. Lönnblad, S. Prestel, Matching Tree-Level Matrix Elements with Interleaved Showers, JHEP 03 (2012) 019.

[92] L. Lönnblad, S. Prestel, Merging Multi-leg NLO Matrix Elements with Parton Showers, JHEP 03 (2013) 166.

[93] L. Lönnblad, S. Prestel, Unitarising Matrix Element + Parton Shower merging, JHEP 02 (2013) 094.

[94] S. Alioli, P. Nason, C. Oleari, E. Re, A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX, JHEP 06 (2010) 043.

[95] W. T. Giele, D. A. Kosower, P. Z. Skands, A simple shower and matching algorithm, Phys. Rev. D78 (2008) 014026.

[96] S. Höche, S. Prestel, The midpoint between dipole and parton showers, Eur. Phys. J. C75 (9) (2015) 461.

[97] J. R. Christiansen, T. Sjöstrand, Weak Gauge Boson Radiation in Parton Showers, JHEP 04 (2014) 115.

[98] L. Carloni, J. Rathsman, T. Sjöstrand, Discerning Secluded Sector gauge structures, JHEP 04 (2011) 091.

[99] B. Andersson, G. Gustafson, B. Nilsson-Almqvist, A Model for Low p(t) Hadronic Reactions, with Generalizations to Hadron - Nucleus and Nucleus-Nucleus Collisions, Nucl. Phys. B281 (1987) 289–309.

[100] B. Nilsson-Almqvist, E. Stenlund, Interactions Between Hadrons and Nuclei: The Lund Monte Carlo, Fritiof Version 1.6, Comput. Phys. Commun. 43 (1987) 387.

[101] C. Bierlich, G. Gustafson, L. Lönnblad, H. Shah, The Angantyr model for Heavy-Ion Collisions in PYTHIA8, JHEP 10 (2018) 134.

[102] S. Ferreres-Solé, T. Sjöstrand, The space-time structure of hadronization in the Lund model, Eur. Phys. J. C78 (11) (2018) 983.

[103] P. Skands, S. Carrazza, J. Rojo, Tuning PYTHIA 8.1: the Monash 2013 Tune, Eur. Phys. J. C74 (8) (2014) 3024.