CERN-TH.6275/91

Monte Carlo Event Generation for LHC*

T. Sjöstrand

CERN — Geneva

Abstract

The necessity of event generators for LHC physics studies is illustrated, and the Monte Carlo approach is outlined. A survey is presented of existing event generators, followed by a more detailed study of the different components that appear in a complete program.

*To appear in the proceedings of the CERN School of Computing, Ystad, Sweden, 23 August — 2 September 1991

CERN-TH.6275/91
October 1991

1 Introduction

A typical event at LHC energies is expected to consist of about 100 charged particles and as many neutral ones, and interesting events often have even higher multiplicities. At the nominal LHC luminosity, every bunch crossing, 15 ns apart, will produce around 20 such events. Many signals for new physics make use of collective properties of these particles, such as jets or missing transverse momentum (from neutrinos and other undetected particles). Others are based on single particles, such as electrons, muons or photons, which then have to be identified with some degree of confidence.

Almost all of the intended signals have their potential backgrounds, wherein everyday processes occasionally have fluctuations that make them look more exotic than they really are. For instance, an isolated high-momentum electron is a main signal for a number of rare processes. On the other hand, an electron produced in a b quark jet is only very rarely isolated, but the total b jet production rate is so huge that even a small fraction of it can be dangerous. A number of detailed questions therefore need to be answered: how well is an electron coming from the signal process really isolated?; how often do possible backgrounds give isolated electrons?; which calorimeter granularity is required to do the job?; and so on.

The task of event generators is to describe, as accurately as possible, the experimental characteristics of physics processes of interest. The main applications are as follows.

- To give physicists a feeling for the kind of events one may expect/hope to find, and at what rates.
- As a help in the planning of a new detector, so that detector performance is optimized, within other constraints, for the study of interesting physics scenarios.
- As a tool for devising the analysis strategies that should be used on real data, so that signal-to-background conditions are optimized.
- As a method for estimating detector acceptance corrections that have to be applied to raw data, in order to extract the 'true' physics signal.
- As a convenient framework within which to interpret the significance of observed phenomena in terms of a more fundamental underlying theory (usually the standard model).

To write a good event generator is an art, not an exact science. It is essential not to trust blindly the results of any single event generator, but always to have several cross-checks. Further, an event generator cannot be thought of as an all-powerful oracle, able to give intelligent answers to ill-posed questions; sound judgement and some understanding of the generator are necessary prerequisites for successful use. In spite of these limitations, the event generator approach is the most powerful tool at our disposal if we wish to gain a detailed and realistic understanding of physics at the LHC before the day when real data are available.

The necessary input for event generators comes from the calculated matrix elements for the different processes, from the measured and parametrized structure functions, from models for parton showers, underlying events and fragmentation, etc., which will be discussed further below.

As the name indicates, the output of an event generator should be in the form of 'events', with the same average behaviour and the same fluctuations as real data. In generators, Monte Carlo techniques are used to select all relevant variables according to the desired probability distributions. The Monte Carlo approach ensures that the proper amount of randomness is included. Normally an 'event' is a list of all final state observable particles, i.e. hadrons, leptons, and photons, together with their momenta. The 'event' thus corresponds to what could actually be seen by an ideal detector. However, often one is only interested in the total energy and direction of a jet, rather than the detailed jet structure. Then a more crude event description, in terms of partons ($\approx$ jets) and leptons, may be enough.

In principle, one must distinguish between an event generator and a numerical integration package for cross-sections: both can be used to evaluate the cross-section for a given process and for given cuts, but only the former gives the full multi-dimensional differential distribution of events within these cuts. In practice, this distinction is not always obvious for a large number of dedicated programs written to study one or a few specific processes: although the main application may be cross-section integration, only little additional effort is needed to generate simple 'events' which consist of a small number of outgoing partons and leptons. At the other end of the generator spectrum, there are large subroutine packages intended for general-purpose use, with many different processes included, and a full description of the production of all hadrons in an event. These packages contain many man-years of effort, 10–30 kilo-lines of Fortran code, and are generally better documented and supported than the smaller packages. Although they may not be the best for all applications, it is natural that we concentrate on them in the following.

Where does a generator fit into the overall analysis chain? In 'real life', the machine produces interactions. These events are observed by detectors, and interesting ones written to tape by the data acquisition system. Afterwards the events may be reconstructed, i.e. the electronics signals (from wire chambers, calorimeters, and all the rest) may be translated into a deduded setup of charged tracks or neutral energy depositions, in the best of worlds with full knowledge on momenta and particle species. Based on this cleaned-up information one may proceed with the physics analysis. In the Monte Carlo world, the rôle of the machine, namely to produce events, is taken by the event generators described in this report. The behaviour of the detectors – how particles produced by the event generator traverse the detector, spiral in magnetic fields, shower in calorimeters, or sneak out through cracks, etc. – is simulated in programs such as GEANT [1]. Traditionally, this latter activity is called event simulation, which is somewhat unfortunate since the same phrase could equally well be applied to what we here call event generation. A more appropriate term is detector simulation. Ideally, the output of this simulation has exactly the same format as the real data registered by the detector, and can therefore be put through the same event reconstruction and physics analysis chain, except that here we know what the 'right answer' should be, and so can see how well we are doing.

Since the full chain of detector simulation and event reconstruction is very time-consuming, often one does 'quick and dirty' studies in which these steps are skipped entirely, or at least replaced by very simplified procedures which only take into account the geometric acceptance of the detector and other trivial effects. One may then use the output of the event generator directly in the physics studies.

Programs still undergo rapid evolution: new processes are calculated and included; improved structure function parametrizations appear; aspects of parton showering, fragmentation and decay are gradually better modelled; and even the physics landscape changes, e.g., as a function of the currently favoured value for the top quark mass. The programs that will be used at LHC are likely to look rather different from those available today. However, many of the basic principles should remain more or less unchanged.

One may also find discussions of, and comparisons between, event generators in many of the recent studies on LHC and SSC physics, such as [2, 3, 4, 5], and in the article [6]. Some of the text for the current article has been borrowed from [6, 7].

2 Monte Carlo Techniques

Quantum mechanics introduces a concept of randomness in the behaviour of physical processes. The virtue of event generators is that this randomness can be simulated by the use of Monte Carlo techniques. The authors have to use a lot of ingenuity to find the most efficient way to simulate an assumed probability distribution. A detailed description of the techniques would carry too far, but for the continued discussion some examples may be helpful.

First of all one assumes the existence of a random number generator. This is a function which, each time it is called, returns a number R in the range between 0 and 1, such that the inclusive distribution of numbers R is evenly distributed in the range, and such that different numbers R are uncorrelated. It is not so trivial to define exactly what one means by 'uncorrelated', or how to ensure that the random number generator indeed produces as uncorrelated numbers as possible. Progress has been made in this area in recent years, however, and simple algorithms with very good properties are now in general use, see [8].

Let us now assume we have a function $f(x)$, with an allowed x range $x_{min} \leq x \leq x_{max}$, and with $f(x)$ non-negative in this range. We want to select an x 'at random' such that the probability for a given x is proportional to $f(x)$. One does not have to assume that the integral of $f(x)$ is normalized to unity; rather, the integral usually forms part of an overall weight factor we want to keep track of; however, that aspect is not covered further here. If it is possible to find a primitive function $F(x)$ which has a known inverse $F^{-1}(x)$, an x can be found as follows (method 1):

$$\int_{x_{min}}^{x} f(x)\,dx = R \int_{x_{min}}^{x_{max}} f(x)\,dx$$
$$\Rightarrow \quad x = F^{-1}(F(x_{min}) + R(F(x_{max}) - F(x_{min}))). \tag{1}$$

However, usually this is not the case. In an alternative method one therefore assumes that the maximum of $f(x)$ is known, $f(x) \leq f_{max}$ in the x range considered. The following scheme will then yield the correct answer (method 2):

1. select an x evenly in the allowed range, i.e. $x = x_{min} + R(x_{max} - x_{min})$;
2. select an h evenly between 0 and f_{max}, i.e. $h = Rf_{max}$ (remember that, although R was also used in point 1, the rules of the game are that anytime R appears it means a new random number);
3. if $h \geq f(x)$ the x value is rejected, and it is necessary to return to point 1 for a new try;
4. else one is done, and the most recent x value is retained as final answer.

Of course, in real life there are a number of complications. On the one hand, x may be a multidimensional vector with complicated boundaries for the allowed region. On the other hand, the function $f(x)$ may be rapidly varying. A common example here is a function with a singularity just outside the allowed region, such that the second method is very inefficient. Alternatively one may have an integrable singularity just at the boundary, and then the method does not work at all. In some cases it may even be difficult to know what is the appropriate f_{max} to use, since the function may have several local maxima in positions not known beforehand. Of course, the method works for any f_{max} bigger than the true maximum, but if one picks f_{max} unnecessarily large, one pays a price in terms of efficiency.

No single method is enough to solve all the conceivable cases. Here we just illustrate three common techniques among the many possible — more may be found e.g. in [9].

Variable transformations may be used to make a function more smooth. Thus a function $f(x)$ which blows up as $1/x$ for $x \to 0$, with an x_{min} close to 0, would instead be roughly constant if transformed to the variable $y = \ln x$. Variable transformations are also often useful to simplify the shape of the boundary of the allowed x region.

Special tricks can sometimes be found. Consider e.g. the generation of a Gaussian $f(x) = \exp(-x^2)$. This function is not integrable, but if we combine it with the same Gaussian distribution of a second variable y it is possible to transform to polar coordinates

$$f(x)\,dx\,f(y)\,dy = \exp(-x^2 - y^2)\,dx\,dy = r\exp(-r^2)\,dr\,d\phi, \tag{2}$$

and now the r and ϕ distributions may be easily generated and recombined to yield x (and a second number y, if one so desires).

Finally, a less straightforward but very useful approach. Assume that we can find a function $g(x) = \sum_i g_i(x)$, such that $f(x) \leq g(x)$ over the x range considered, and such that the functions $g_i(x)$ each are simple in the sense that we can find primitive functions and their inverses. In that case (method 3):

1. select an i at random, with relative probability given by the integrals

$$\int_{x_{min}}^{x_{max}} g_i(x)\,dx = G_i(x_{max}) - G_i(x_{min}); \tag{3}$$

2. for the i selected, use method 1 to find an x, i.e.

$$x = G_i^{-1}(G_i(x_{min}) + R(G_i(x_{max}) - G_i(x_{min}))); \tag{4}$$

3. select an $h = Rg(x)$;
4. if $h \geq f(x)$ reject the x value and return to point 1;
5. else one is done.

For a function $f(x)$ which is known to have sharp peaks in a few different places, the generic behaviour at each peak separately may be covered by one or a few simple functions $g_i(x)$, to which one adds a few more $g_i(x)$ to cover the basic behaviour away from the peaks. By suitable selection of the relative strengths of the different g_i:s (which may be left for the computer to figure out at an initialization stage), it is possible to find a function $g(x)$ which is matching well the general behaviour of $f(x)$, and thus to achieve a high Monte Carlo efficiency.

3 Overview of Event Generators

The perfect event generator does not exist. This reflects the limited understanding of physics in many areas. Indeed, a perfect generator can only be constructed once everything is already known, in which case experiments are superfluous. One therefore has to be satisfied with programs which are in reasonable agreement with already accumulated experience, theoretical and experimental, and which provide sensible extrapolations to higher energies. Since the ultimate goal is to look for new physics, it is also necessary to include the simulation of different alternative scenarios.

Given the complexity of the problem, the Monte Carlo approach allows a convenient division into separate subtasks. Thus, to describe an event in full, one needs to consider the following components:

1. The hard scattering matrix elements. These define the process(es) under study, and are therefore at the core of the programs.
2. The structure functions. The differential cross-sections, which are to be simulated in the programs, are given as the products of structure functions and the hard scattering matrix elements above.
3. Final state radiation. Partons in the final state may radiate. At high energies, this perturbative radiation is the dominant mechanism for building up the structure of (high-$p_\perp$) jets, with broad jet profiles and subjets.
4. Initial state radiation. The incoming partons may also radiate before the hard interaction, thus giving rise to additional jets close to the directions of the incoming hadrons.
5. Beam jets. Only one parton from each incoming hadron is assumed to participate in the hard interaction, and in the initial state showering. All the other partons act to produce the beam jets found along the directions of the original incoming hadrons.
6. Fragmentation and decays. Partons are not directly observable. Instead, once sufficiently removed from each other, they are fragmented into a collection of hadrons. Many of these hadrons are unstable, and subsequently decay.

Of course, this separation is very crude and schematic. Thus, one and the same $2 \to 3$ process might be described either in terms of a basic $2 \to 3$ matrix element, or in terms of a $2 \to 2$ hard scattering followed by final state radiation, or in terms of a $2 \to 2$ hard scattering preceded by initial state radiation. It is therefore important to join the different descriptions in a consistent manner, e.g. to avoid double counting.

The double counting issue is non-trivial, and in practice it has led to a split of the Monte Carlo program activity into two different approaches, which we will refer to as 'parton showers' (PS) and 'matrix elements' (ME), respectively.

In the parton shower approach, it is customary to implement only the lowest order matrix elements, i.e. as a rule, basic $2 \to 2$ processes. Initial and final state radiation are added on to the basic scattering in the shower approach proper. The showers are assumed to be universal, i.e. the shower evolution is not allowed to depend on the details of the hard scattering, but only on the gross features: energies and flavours of incoming and outgoing partons, and an overall Q^2 scale for the hard scattering. The approximate nature is reflected in a limited accuracy for the rate of production of additional well-separated jets, but the internal structure of jets should be well modelled. It is feasible to add fragmentation and beam jets, and thus to generate realistic representations of the events produced in hadron colliders. In this category of programs, a large fraction of the total investment is in the common shower and fragmentation routines, while the effort needed to include yet another $2 \to 2$ process is modest, if only matrix elements are known and not too complex. Some of the programs of this kind therefore allow the simulation of many different processes.

The list of such event generators is fairly small. We are aware of the following programs:

- ISAJET, by Paige and Protopopescu, current version 6.36 [10].
- PYTHIA, by Bengtsson and Sjöstrand, current version 5.5 [11].
- HERWIG, by Marchesini and Webber, current version 5.3 [12].
- COJETS, by Odorico, current version 6.11 [13].
- DTUJET, by Ranft *et al.* [14].
- FIELDAJET, by Field *et al.* [15].
- The Fire-String program by Angelini *et al.* [16].
- FRITIOF, by Andersson *et al.*, current version 6.0 [17].

Without passing judgement on quality, the ordering above does reflect an element of quantity: ISAJET and PYTHIA are clearly more versatile than the others, with HERWIG up-and-coming, while the latter four programs only cover QCD jets and minimum bias events.

The matrix element approach is represented by another class of programs. Here the emphasis is on the use of exact higher-order matrix elements. The analytic formulae in the programs are considerably more complicated, and the phase space generation machinery more advanced. The big investment here is in the matrix element calculation itself — usually these programs are written by the same people who calculated the matrix elements in the first place — and in selecting the kinematic variables in an efficient way. There is therefore less impetus for a common approach to many disparate processes. Since the precision aspect is important, it is not feasible to attach a simple, generic parton shower picture. Normally, therefore, only a fixed (small) number of partons are generated. Since most modern fragmentation models are tuned to be attached at the end of the parton shower evolution, fragmentation and beam jet treatments also become less interesting. These programs therefore mainly generate parton configurations of 'pencil jets', rather than events as they may appear in a detector.

The number of matrix element programs is considerably higher than the number of parton shower programs: once a matrix element has been calculated, the Monte Carlo approach is usually the most convenient way to obtain physical cross-sections. Therefore many calculations are directly turned into programs. It is not possible in this report to give a complete list of all programs of this kind, some of which are publicly maintained and others which are not. Two programs contain matrix elements for widely different purposes:

- PAPAGENO, by Hinchliffe [18].
- EUROJET, by van Eijk *et al.* [19].

A few others will be mentioned in connection with the processes they simulate.

The parton shower and matrix element programs fill somewhat complementary functions. The former are convenient for exploratory work: it is fairly easy to simulate a new, postulated physics process in sufficient detail to establish experimental feasibility, and to try out the tools needed to separate signal from background. For high-precision measurements of an established process, on the other hand, one needs the higher order matrix elements. The matrix element programs are also more convenient for generating events within very specific phase space regions, since the cuts can be included from the start. With parton shower based programs it is necessary to generate more inclusive event samples and afterwards discard those events that do not fulfil the requirements, a procedure which can often be very inefficient.

3.1 Kinematics and cross-sections

In this section we describe how the hard scattering process is generated. The example is for the case of a $2 \to 2$ process, with mass effects neglected, but the same basic principles apply also for other cases.

Consider two incoming protons in their CM frame, each with energy E_{beam}. The total CM energy-squared is then $s = 4E_{beam}^2$. The two partons that enter the hard interaction do not carry the total beam momentum, but only fractions x_1 and x_2, respectively, i.e. they have four-momenta

$$\begin{aligned} p_1 &= E_{beam}(x_1; 0, 0, x_1), \\ p_2 &= E_{beam}(x_2; 0, 0, -x_2). \end{aligned} \tag{5}$$

The invariant mass-squared of the two partons is defined as

$$\hat{s} = (p_1 + p_2)^2 = x_1 x_2 s. \tag{6}$$

Instead of x_1 and x_2, it is often customary to use τ and either y or x_F:

$$\tau = x_1 x_2 = \frac{\hat{s}}{s}; \tag{7}$$

$$y = \frac{1}{2} \ln \frac{x_1}{x_2}; \tag{8}$$

$$x_F = x_1 - x_2. \tag{9}$$

In addition to x_1 and x_2, two additional variables are needed to describe the kinematics of a scattering $1 + 2 \to 3 + 4$. One corresponds to the azimuthal angle ϕ of the scattering plane around the pp axis. This angle is isotropically distributed (unless the protons are polarized), and so need not be considered further. The other variable could have been picked as $\hat{\theta}$, the polar angle of parton 3 in the CM frame of the hard scattering. However, the conventional choice is to use the variable

$$\hat{t} = (p_1 - p_3)^2 = (p_2 - p_4)^2 = -\frac{\hat{s}}{2}(1 - \cos\hat{\theta}), \tag{10}$$

with $\hat{\theta}$ defined as above. It is also customary to define $\hat{u}$,

$$\hat{u} = (p_1 - p_4)^2 = (p_2 - p_3)^2 = -\frac{\hat{s}}{2}(1 + \cos\hat{\theta}), \tag{11}$$

but $\hat{u}$ is not an independent variable since

$$\hat{s} + \hat{t} + \hat{u} = 0. \tag{12}$$

The cross-section for the process $1 + 2 \to 3 + 4$ may be written as

$$\begin{aligned}\sigma &= \int\int\int dx_1\, dx_2\, d\hat{t}\, f_1(x_1, Q^2)\, f_2(x_2, Q^2)\, \frac{d\hat{\sigma}}{d\hat{t}} \\ &= \int\int\int \frac{d\tau}{\tau}\, dy\, d\hat{t}\, x_1 f_1(x_1, Q^2)\, x_2 f_2(x_2, Q^2)\, \frac{d\hat{\sigma}}{d\hat{t}}. \end{aligned} \quad (13)$$

The $f_i(x, Q^2)$ are the structure functions of the proton, which express the probability to find a parton of species i with a momentum fraction x inside a proton probed at a virtuality scale Q^2. The choice of Q^2 scale is ambiguous; one common alternative is the transverse momentum-squared

$$Q^2 = p_\perp^2 = \frac{\hat{s}}{4}\sin^2\hat{\theta} = \frac{\hat{t}\hat{u}}{\hat{s}}. \quad (14)$$

The $d\hat{\sigma}/d\hat{t}$ expresses the differential cross-section for a scattering, as a function of the kinematical quantities $\hat{s}$, $\hat{t}$ and $\hat{u}$. It is in this function that the physics of a given process resides.

The performance of a machine is measured in terms of its luminosity $\mathcal{L}$, which is directly proportional to the number of particles in each bunch and to the bunch crossing frequency, and inversely proportional to the area of the bunches at the collision point. For a process with a σ as given by eq. (13), the differential event rate is given by $\sigma\mathcal{L}$, and the number of events collected over a given period of time

$$N = \sigma \int \mathcal{L}\, dt. \quad (15)$$

3.2 Hard scattering subprocesses

Lists of subprocesses included in Monte Carlos are found in Tables 1 and 2. These tables should be read as follows. For ISAJET, PYTHIA and PAPAGENO, a '•' indicates that the process is included and a '-' that it is not. In the column 'other PS' (PS = parton shower programs) a '•' indicates this is something found in most or all programs in this category, while an 'H' appears if only HERWIG includes it and a blank if no program does. In the column 'other ME' (ME = matrix element programs), an 'E' indicates a process included in EUROJET, and other letters indicate processes found in other programs, as explained further in the process-specific descriptions below.

The tables should be taken as indicative only, since there is a continuous evolution of many programs. Furthermore, one and the same process may be treated differently in different programs. Below we will give some comments on a few of the processes, to illustrate the degrees of freedom open to Monte Carlo authors.

3.2.1 QCD

Exact Born term cross-sections, for up to five jets in the final state, are available in the NJETS program of Kuijf and Berends ('NJ' of Table 1), see [20], which is the most advanced in this category. This program also contains approximate expressions for up to eight jets.

Complete loop calculations have been performed up to $\mathcal{O}(\alpha_S^3)$. These are implemented in the numerical integration programs of two groups [21], but no event generators exist so far.

Most programs only contain the lowest order Born term cross-sections for heavy flavour production. For top this may be sufficient, i.e. higher order contributions effectively contribute an overall K factor, but do not significantly change the production characteristics

Table 1: Standard model physics processes included in the event generators studied. See text for program notation. 'f' stands for fermion, 'V' for W or Z, and 'Q' for heavy quark.

Process	ISAJET	PYTHIA	other PS	PAPA-GENO	other ME
QCD					
QCD jets	•	•	•	•	E, NJ
$q\overline{q}, gg \to t\overline{t}$	•	•	•	•	E
$qb \to q't$	-	•	H	-	
$J/\Psi/\Upsilon$ (direct)	-	-		-	
minimum bias	•	•	•	-	E
diffractive	•	•		-	
elastic	-	•		-	
Prompt photons					
$qg \to q\gamma, q\overline{q} \to g\gamma$	•	•	H	•	
$q\overline{q} \to \gamma\gamma$	•	•		•	
$gg \to \gamma\gamma$	-	•		•	
W/Z production					
$q\overline{q} \to V$	•	•	•	•	LD
$qg, q\overline{q} \to V(q,g)$	•	•	H	•	LD
$q\overline{q} \to VV, V\gamma$	•	•		•	BZ, BH
$q\overline{q}, gg \to VV(q,g)$	-	-		-	VV, BH
$gg \to VV, V\gamma$	-	-		-	GG
$q\overline{q} \to V^*V^*$	-	•		-	
$gg \to ZQ\overline{Q}$	-	•		•	LD
Standard model H^0 ($m_H \leq 800$ GeV)					
$q\overline{q} \to H^0$	•	•		-	
$gg \to H^0$	•	•	H	•	GG
$VV \to H^0$	•	•	H	•	BG
$q\overline{q} \to VH^0$	-	•		-	
$gg, qg, q\overline{q} \to H^0(q,g)$	-	•		-	HV
$H^0 \to VV$	•	•	H	•	BG, HV
$H^0 \to V^*V^*$	•	•	H	-	
$H^0 \to f\overline{f}$	•	•	H	-	
$H^0 \to gg$	•	•		-	
$H^0 \to \gamma\gamma$	•	•	H	-	
$H^0 \to \gamma Z^0$	-	•		-	
Standard model H^0 ($m_H \geq 700$ GeV)					
$VV \to VV$	•	•	H	-	BG
$gg \to VV$	-	-		-	GG

Table 2: Non-standard model physics processes included in the event generators studied. See text for program notation. In addition to notation for Table 1, 'V'' stands for W' or Z', 'R' for a horizontal boson, and 'L' for heavy lepton.

Process	ISAJET	PYTHIA	other PS	PAPA-GENO	other ME
Non-standard Higgs particles					
h, H, A as above	-	•	H	-	
$Z^* \to h^0 A^0$, $H^0 A^0$	-	•		-	
$q\overline{q}' \to H^+$	-	•		-	
$gb \to H^- t$	-	•		-	
$\gamma^*/Z^* \to H^+ H^-$	-	•		-	
$t \to H^+ b$	-	•		-	
$H^+ \to f\overline{f}'$	-	•		-	
Supersymmetry					
$q\overline{q}, gg \to \tilde{q}\tilde{\overline{q}}$	•	-		•	UA, BT
$q\overline{q}, gg \to \tilde{g}\tilde{g}$	•	-		•	UA, BT
$qg \to \tilde{q}\tilde{g}$	•	-		•	UA, BT
$q\overline{q} \to \tilde{g}\tilde{V}$	•	-		-	UA
$qg \to \tilde{q}\tilde{V}$	•	-		-	UA
$\tilde{q}, \tilde{g}, \tilde{V}$ decays	-	-		•	UA, BT
New Gauge Groups					
$q\overline{q} \to V'$	-	•		-	
$VV \to V'$	-	-		-	
$V' \to f\overline{f}$	-	•		-	
$V' \to VV$	-	•		-	
$q\overline{q}' \to R \to q''\overline{q}'''$	-	•		-	
Fourth Generation					
$q\overline{q}, gg \to Q\overline{Q}$	•	•		•	
$V/V' \to Q\overline{Q}, L\overline{L}$	-	•		-	
$qq' \to q''Q$	-	•		-	
Other Topics					
contact interactions	-	•		•	
axigluons	-	-		•	
leptoquarks	-	•		-	E
strongly interacting V	-	•		-	
q^* (excited fermions)	-	•		-	

of top. However, at LHC energies, it is not correct to use only the Born term to estimate b or c production, since these quarks receive major higher order contributions, both by flavour excitation and by parton shower evolutions.

Minimum bias physics is discussed together with beam jets below.

Even when diffractive and elastic scattering is included in programs, the treatment is fairly primitive, and likely to be insufficient for LHC physics. Several major features are missing, like high-$p_\perp$ jet production in diffractive events [22].

3.2.2 Prompt photons

Complete next-to-leading order programs for prompt photon production are available from two groups [23], but both are intended for cross-section calculation rather than event generation. Leading order formulae are contained in many event generators. Some parton shower algorithms also include the emission of photons as part of the evolution.

The $gg \to \gamma\gamma$ graph contains a quark box. The cross-section is reasonably compact in the limit of vanishing quark mass, but very complex if the correct quark mass dependence is included. Therefore often the massless formulae are used, with the number of flavours suitably chosen. PYTHIA contains the full formulae as an option, but these then are numerically unstable in some regions of phase space, and therefore not easy to use.

3.2.3 W/Z production

The most complete W/Z ME program is the 'Leiden-Durham W'/VECBOS program ('LD' of Table 1), which contains the production of a V, i.e. a W or a Z, plus 0, 1, 2, 3 or 4 jets, see [24]. No loop corrections are available in this program, but analytical formulae exist up to second order in α_S [25]. Programs for the production of $V+V$ and $V+\gamma$ are also available, in [26] ('BZ' of Table 1) with special emphasis on the possibility of testing for anomalous couplings in triple gauge boson vertices. The production of a VV pair plus one additional jet is found in two programs: in VVJET [27] ('VV' of Table 1) and in [28] ('BH' of Table 1); the latter also contains matrix elements for a VV pair plus two jets. In all the programs above, subsequent V decays are included, with full angular correlations.

As in $\gamma\gamma$ pair production, VV pairs may also be produced from a gg initial state, via a quark box. The rates may be sizeable, thanks to the large value of the gluon structure functions at the small x values probed by LHC, and interference with the Higgs signal is of particular importance for Higgs searches. The program GGZZ simulates this process [29] ('GG' of Table 1).

The parton shower programs tend to give a fairly good description of V production at current energies. However, the rate of high-$p_\perp$ V production is not so well reproduced if the starting point is the $q\overline{q} \to V$ matrix element. One may instead use the $qg \to Vq$ and $q\overline{q} \to Vg$ matrix elements, in which case at least one high-$p_\perp$ jet is assured from the start, and then include showering to generate additional jets. This gives a better description at high $p_{\perp V}$, but cannot be used to describe inclusive V production, since the $2 \to 2$ matrix elements are divergent for $p_\perp \to 0$. The choice between the two descriptions therefore has to depend on the application. In ISAJET a special option is available, in which the $2 \to 2$ matrix elements have been regularized (by hand) in the limit $p_\perp \to 0$, and so a good description is obtainable over the whole $p_\perp$ spectrum.

For intermediate mass Higgs background studies, the $Z + (Z^*/\gamma^*)$ (where * denotes that the interesting configurations are those with the particle far off mass-shell) and $Zb\bar{b}$ channels are of particular interest. The latter process is calculated in [30], and is now included in a few generators, although still with an inefficient selection of phase-space points.

3.2.4 Standard model H^0

A single unified description of Higgs production and decay characteristics, valid for all Higgs masses, would be very complex. In practice, two different descriptions are in use in programs. For a reasonably light Higgs, and thereby a reasonably narrow one, the 'signal' and the 'background' graphs do not interfere significantly, so that it is possible to separate the process into Higgs production and Higgs decay. If the Higgs is heavy, this is no longer possible but, in this region, mainly the $VV \to H \to VV$ graphs are of experimental interest, and so only full interference with the $VV \to VV$ background need be included.

A light or intermediate mass Higgs is predominantly produced by $gg \to H$. The process $VV \to H$, i.e. properly $qq' \to q''q'''H$, also contributes. This process is included with the full matrix elements in PYTHIA, HERWIG and PAPAGENO. In ISAJET the effective W approximation is used, which is known to be good for $m_H \gg m_W$, but not so good for lower Higgs masses.

In the description of Higgs decays, two new aspects have played a particular rôle in recent activity. One is the introduction of running quark masses for couplings $H \to q\overline{q}$; this typically leads to a reduction of the quark partial widths by a factor of around 2. At intermediate Higgs masses, where the $H \to b\bar{b}$ decays dominate, some other branching ratios are enhanced by the same factor 2, notably $H \to \gamma\gamma$. Running quark masses are included in PYTHIA and HERWIG, but not in ISAJET. The other new aspect is $H \to V^{(*)}V^*$ decays, i.e. where one or both final state gauge bosons are significantly off mass-shell. Particularly interesting are the $H \to ZZ^* \to 4\ell$ decays, which now are found in ISAJET, PYTHIA and HERWIG.

For the heavy Higgs scenario, both ISAJET and PYTHIA rely fully on the effective W approximation for $VV \to VV$ matrix elements. In both programs the incoming V bosons are assumed longitudinally polarized, as are the outgoing in PYTHIA, while ISAJET includes all polarization combinations in the final state. A more detailed description, based on exact matrix elements with full interference between all graphs that can yield VV plus two jets in the final state, is found in [31] ('BG' of Table 1); full angular correlations in the V decays are also included.

Finally, just as for the description of high-$p_\perp$ V production, it may be convenient to have a description of a H recoiling against a jet; this is available in the program HVVJET [32] ('HV' of Table 1).

3.2.5 Non-standard Higgs particles

Little effort has gone into scenarios with more Higgses than in the standard model. However, recently many of the production processes in the two-Higgs-doublet scenario of the Minimal Supersymmetric Standard Model (MSSM) were included in PYTHIA, and steps in the same direction have been taken in HERWIG. In the MSSM, there are five physical Higgses: two neutral scalars h^0 and H^0, one neutral pseudoscalar A^0, and two charged particles $H^\pm$. The production of the neutral particles follows the same pattern as that of the standard model H^0, except that couplings are changed, and that pair production e.g. of $h^0 + A^0$ become of interest. It also becomes necessary to cover the possibility of sequential decays of one Higgs state into another, e.g. by Z^0 emission. The charged Higgs may be produced singly or in pairs; one potentially significant source is top decays.

3.2.6 Supersymmetry

SUSY is an important area to be explored at LHC. Several different particles should be searched for, in particular squarks, gluinos and a host of gauginos. Two main programs in this area are ISAJET and UA2SUSY. As the name indicates, the latter ('UA' of Table 2) is an upgrade of a dedicated program written inside the UA2 collaboration [33]. A further program is found in [34] ('BT' of Table 2). Recent developments include a special emphasis on a flexible and detailed modelling of all sequential decay chains predicted for different parameter sets of the MSSM.

3.2.7 New gauge groups

A number of different scenarios can give rise to new gauge particles, here denoted V' ($= Z'^0$ or $W'^{\pm}$). In PYTHIA, vector and axial couplings of fermions to the V' have been left as free parameters; it is therefore possible to simulate most of the alternatives on the market by judicious choices. Couplings of a V' to the standard model gauge bosons can show a richer structure, and only a few of the possibilities are available here.

A specific model for a horizontal boson R, i.e. a boson which couples to generation number, has been included as a separate alternative in PYTHIA.

3.2.8 Fourth generation

With the current LEP limits on the number of light neutrino species, the prospects are slim for a standard fourth generation of fermions. Should there still be some interest in heavy standard quarks or leptons, the event generators are available, since only trivial extensions of the standard description of top are involved.

3.2.9 Other topics

The list of possible extensions to, or deviations from, the standard model is long, and only a few are found in Table 2. Among the most interesting ones are the prospects of a strongly interacting V sector, as could arise if the standard model Higgs were absent or, at least, much heavier than the 1 TeV mass scale directly probed. Some of the scenarios proposed in the literature have been implemented in PYTHIA.

3.3 Structure Functions

The proton is not a static object. In addition to the three valence quarks, virtual gluons and quark-antiquark pairs are continually created and annihilated. Currently a first-principles complete picture does not exist — maybe lattice QCD studies one day will provide that. Meanwhile, one makes do with a simple probabilistic picture, where the structure functions $f_i(x, Q^2)$ give the probability to find a parton i with fraction x of the proton momentum, if the proton is probed at a scale Q^2. As the scale is increased, more and more short-lived fluctuations can be probed. Therefore the structure functions change in a characteristic fashion expressed by the evolution equations

$$\frac{\partial f_i(x, Q^2)}{\partial(\ln Q^2)} = \sum_j \int_x^1 \frac{dx'}{x'} f_j(x', Q^2) \frac{\alpha_S(Q^2)}{2\pi} P_{j\to i}\left(\frac{x}{x'}\right), \tag{16}$$

where α_S is the strong coupling constant. The splitting kernels $P_{j\to i}$ are described further below, eq. (18). While thus the Q^2 dependence is predicted by theory, it is necessary to determine the functions at some fixed scale Q_0^2 by comparison with data, in particular from deep inelastic scattering experiments.

A community of people is involved in the analysis of data and the extrapolation to unmeasured regions, based on the QCD evolution equations. The end result of these efforts is new structure function sets, with some region of validity in the (x, Q^2) plane. In the past, the number of sets available was fairly limited; for applications at the large Q^2 scales of LHC/SSC, only the EHLQ parametrizations [35] could be used, which is why these are still found as defaults in many programs.

More recently, the pace has picked up, and now new sets appear almost monthly. A review of, and comparison between, most of these is found in [36]. One conclusion is that many of the older sets do not do well when compared with current data, and therefore should no longer be used. Also some of the newer sets perform less well. In part, this is deliberate: given the large uncertainties involved, most authors do not provide one single 'best' set, but rather prefer to produce many different sets, which together are supposed to bracket the 'right' answer. The differences between these sets come from the correlation between the choice of Λ value (in α_S) and the choice of gluon structure function, from different assumptions about the behaviour of structure functions at low x, from different choices of strange quark distributions at low Q^2, etc.

Since all sets of structure functions are limited in validity to given x and Q^2 ranges (in particular, $x > 10^{-5}$ to 10^{-4}, depending on the set), their use for applications at LHC/SSC energies need extra care. Cross-sections and differential distributions (e.g. for b or c quark production) could be affected. To overcome the problem in part, Monte Carlo authors may have to introduce further assumptions themselves.

An additional element of disparity comes from the choice of order and renormalization scheme. The three main alternatives are leading order, next-to-leading order in the $\overline{\mathrm{MS}}$ scheme, and next-to-leading order in the DIS scheme. For high precision measurements, it is essential to use the same conventions for matrix elements and structure functions, and here probably little confusion exists. The appropriate choice to use for parton shower based programs may be less clear — while basically leading log, these programs do include some next-to-leading log contributions.

A main programming issue for structure functions is whether to use grids or parametrizations. In the former approach, the output of the evolution programs is stored directly as grids in the (x, Q^2) plane, and desired values can be obtained by interpolation in these grids. The drawback is that thousands of real numbers have to be transferred to each new computer as external files, which makes programs a little less easily transportable. The advantage is that interpolation is usually fast. In the parametrization approach, smooth functions are fitted to the grid values, and subsequent use is based on these fits. This way the number of real values that characterize a structure function set is significantly reduced, hopefully without any loss of information — the most spectacular example is the very compact parametrizations by Morfín and Tung [37]. Such parametrizations can easily be included in the code of an event generator, and thus there are no transport problems. Since the evaluation typically involves logarithms and exponents, it may be significantly slower than in the grid interpolation approach, however.

Recently, a program PDFLIB was released [38], which puts together basically all existing structure functions into one single package, with a common calling structure. This has greatly simplified the task of Monte Carlo authors, who now only have to provide an interface to this library.

3.4 Initial and Final State Showers

In the parton shower approach, a hard $2 \to 2$ scattering is convoluted with initial and final state radiation to build up multiparton final states. Of the two showering types, final state radiation is theoretically and experimentally well under control, while initial state radiation remains less well understood.

As noted above, the shower approach is expected to do a good job for small-angle emission, which is the one that determines the internal structure of jets. It is inferior to the matrix element approach for the rate of well-separated jets but can, to some extent, be tuned to give a reasonable overall description also in this region.

In leading log, the probability $\mathcal{P}$ for branchings $q \to qg$, $g \to gg$, and $g \to q\overline{q}$ is described by the standard evolution equations

$$\frac{d\mathcal{P}}{d(\ln Q^2)} = \int dz \, \frac{\alpha_S(Q^2)}{2\pi} \, P_{a\to bc}(z), \tag{17}$$

where

$$
\begin{aligned}
P_{q\to qg}(z) &= \frac{4}{3}\,\frac{1+z^2}{1-z}, \\
P_{g\to gg}(z) &= 3\,\frac{(1-z(1-z))^2}{z(1-z)}, \\
P_{g\to q\bar{q}}(z) &= \frac{1}{2}(z^2+(1-z)^2). \qquad (18)
\end{aligned}
$$

The z variable describes the sharing of energy (and momentum) between the daughter partons, with parton b taking a fraction z and c the remaining $1-z$ of the original a energy. The probability for soft gluon emission is divergent, and is normally regularized by requiring some minimum parton energy, i.e. $z_{min} \leq z \leq z_{max}$.

Equation (17) describes the probability for a single branching $a \to bc$. Once formed, the daughters b and c may in their turn branch, and so on, so that a tree-like structure develops. The shower evolution is cut off at some minimal scale, typically $Q_0 \approx 1$ GeV. Below this scale, perturbation theory is assumed to break down, and non-perturbative fragmentation takes over.

Although eq. (17) does not seem to distinguish between initial and final state radiation, in fact the difference is quite significant. This will be outlined in the following.

Final state showers are timelike: the two outgoing partons of a $2 \to 2$ scattering each has $m^2 = E^2 - \overline{p}^2 \geq 0$. The evolution is therefore in terms e.g. of $Q^2 = m^2$, and in each successive branching the daughters are constrained by kinematics to have a smaller m^2 than that of their mother.

The naive leading log parton shower picture is modified by coherence effects, which can be taken into account by the inclusion of angular ordering [39], i.e. not only are virtualities successively degraded, but so are the opening angles of branchings. The scale of α_S is also changed from m^2 to $p_\perp^2 \approx z(1-z)m^2$. Further details on the theory of timelike showers may be found in several reviews, e.g. [40, 41].

On the experimental front, final state showers have been much studied in e^+e^- annihilation; since no initial state QCD showers appear in e^+e^-, and since the production graph is s-channel only, the analysis is simpler than in hadron collisions. The recent LEP results underline how well existing showering programs do, see e.g. [42, 43]. It is seldom that disagreements between data and programs like JETSET (which is the program used for showering in PYTHIA) or HERWIG reach the 10% level. Even more importantly, with parameters tuned at LEP, programs also do a good job of describing data at lower energies, at PEP, PETRA and TRISTAN. Confidence in extrapolations to higher energies is therefore high.

Anytime one has to consider the hadronic decay of a colour singlet particle in hadron colliders, such as W, Z, H, etc., the e^+e^- experience is directly applicable, and predictive power high. In principle, questions could be raised whether colour exchange might take place between the partons of the decaying singlet particle and the partons of the underlying event; such effects could modify event topologies, but probably not drastically. When the hard process does not go through a colour singlet intermediate state, on the other hand, there are significant ambiguities in how to begin the shower evolution at high virtualities, such that the proper amount of multijet activity is obtained. Once a choice is made here, the subsequent evolution is again well under control.

Initial state radiation is considerably more difficult to model. The shower is initiated by a parton selected from structure functions at small Q^2. This parton may now branch, but in the branching only one daughter is timelike, whereas the other is spacelike, i.e. $m^2 < 0$. The timelike parton may develop a shower, very much like the final state radiation case, although typically with less allowed phase space and therefore less extensive. The spacelike parton may branch once again, to a new pair of one timelike and one spacelike daughter, etc. The sequence of spacelike daughters is terminated at the hard interaction:

a hard $2 \rightarrow 2$ (QCD) process consists of two incoming spacelike partons and two outgoing timelike ones. In leading log language, the virtuality $Q^2 = -m^2$ of the sequence of spacelike partons is required to increase monotonically, and is constrained from above by the Q^2 scale of the hard interaction.

The inclusive parton distribution at each Q^2 scale is given by structure functions evolved according to eq. (16) — the difference between eq. (16) and eq. (17) is that the latter refers to the branchings of an individual parton while the former considers the evolution of the parton density as a whole. In a Monte Carlo, it is therefore not necessary to perform the initial state evolution before the hard interaction is selected. Rather, one may use already evolved structure functions to select the kinematics of the hard interaction, and only thereafter reconstruct the shower history that preceded this interaction. This is conveniently done in terms of the 'backwards evolution' scheme [44], where the reconstruction of the spacelike evolution is begun at the hard interaction and thereafter gradually carried to earlier and earlier times, until the cut-off scale Q_0 is reached.

In recent years, theoretical progress has been made in including coherence corrections to the leading log picture [45]. The complexity of these corrections is such that no program includes all effects in full, however. HERWIG is the program that contains the most advanced machinery. It has still not been clarified exactly how big the differences are compared to the more simpleminded approaches in other programs.

3.5 Beam Jets

The description of beam jets, i.e. the physics of underlying events and minimum bias events, remains the least well understood aspect of Monte Carlo modelling of hadronic events. It is therefore possible to choose many possible approaches.

The most naive would be to associate each beam remnant with a single jet, which typically would have to contain a leading baryon, but for the rest look like an ordinary quark jet. This approach does not work, in that it gives too low an average charged multiplicity and too narrow a multiplicity distribution compared to data.

One simple way out is to decouple the fragmentation of beam jets from that of ordinary jets. This is done e.g. in COJETS and HERWIG, where the particle multiplicity is selected according to a parametrization of $p\overline{p}$ data, and particles thereafter distributed according to longitudinal phase space, i.e. uniformly in rapidity. The underlying physics is here left unexplained.

An explanation from first principles may be obtained if the composite structure of the proton is invoked, to motivate the possibility that several parton-parton interactions may take place in one and the same proton-proton collision. In ISAJET and DTUJET this is achieved within the Cut Pomeron/Dual Topological Unitarization framework, which uses unitarity arguments to derive a multiplicity distribution in the number of parton-parton interactions per event. Longitudinal momentum sharing between beam partons is based on theoretical low-Q^2 structure functions, while partons do not have any transverse momenta. In PYTHIA the rate of multiple parton-parton interactions is based on the standard perturbative QCD formulae, with an effective cut-off at around $p_{\perp min} \approx 1.6$ GeV, a number that does not come out of any physics analysis but is obtained by a tuning to data. Here partons thus have both longitudinal and transverse momenta.

If, in the end, the programs above agree reasonably well, it is mainly because they have been tuned to the same data, making use of the freedom inherent in all current approaches. However, detailed comparisons between programs and data are scarce.

Theoretical work on the structure of minimum bias events has been carried out in particular by Levin and Ryskin [46]. Their approach is also based on a multiple parton-parton interaction scenario. Compared to the models above, particular emphasis is put on saturation effects at small x. Saturation can arise when the local density of partons becomes so large that not only parton branchings but also parton recombinations have to be taken into account. This saturation is predicted to set in sooner than given by naive estimates, since a large fraction of the partons inside a proton are assumed to be concentrated in a few 'hot spots'. If correct, naive extrapolations to LHC energies, as embodied in current event generators, may fail. Some first hints on the validity of the Levin-Ryskin model may come already with HERA.

3.6 Fragmentation and Decays

Fragmentation is a non-perturbative phenomenon, and as such is not yet understood from first principles. As with timelike parton showers, experience from e^+e^- annihilation helps constrain models significantly [47, 42, 43]. Three different main fragmentation schools exist: string (found in PYTHIA and FRITIOF), cluster (HERWIG) and independent (e.g. ISAJET, COJETS and EUROJET) fragmentation.

In the string and cluster approaches, the colour topology of the event affects the distribution of the final state hadrons. If two partons share a colour-anticolour pair, a string (or set of clusters), is stretched between these partons, such that low-momentum hadrons are produced predominantly in that angular range. In independent fragmentation, each parton fragments on its own, such that soft particles are evenly distributed in azimuth around the respective jet axis. Hard particles do not distinguish between models, but are always produced close to the original parton directions.

The string and cluster models are known to give good agreement with e^+e^- data over a wide range of energies, and are expected to work well also at higher energies. The independent fragmentation model currently is not much used in e^+e^-, and is known to be unable to describe some critical distributions, although many other distributions can again be described well. On theoretical grounds, independent fragmentation is disfavoured due to a number of (conceptually) ugly features. Differences between models are difficult to find in hadron collisions, because here the physics is considerably more smeared out by a variety of effects. However, predictions exist and may one day be tested with sufficient precision. These predictions involve both the colour flow internally between three or more high-$p_\perp$ partons, and between the high-$p_\perp$ partons and the beam remnants.

It should be recognized that not all aspects are fully understood. Consider e.g. $qg \to qg$. In this process there are two possible colour flow topologies, and therefore two possible string configurations. However, the standard QCD matrix elements in fact also contain an interference term between the two, where the colour flow is not well defined. For practical applications, this term is small and may normally be neglected but, ideologically, it is not known how to achieve a fully correct description.

A majority of the particles produced in the fragmentation step are unstable and decay further. Almost all programs therefore include decay routines, more or less similar to each other. Decay data are taken from [48], where available, and according to the best understanding of the program author, where not. There are some differences in level of sophistication, with respect to inclusion of decay matrix elements and polarization information, but seldom does this give readily visible experimental consequences.

4 Programming Issues

4.1 How to Run a Program

Each program has its own style of usage. Some require a user-written main program where variables are set and the program routines called, in others the main program comes with the package and it calls on user-provided subroutines at specific moments, in others still the main program of the package reads a deck of data cards for instructions. The possibilities for the user to modify the basic behaviour are endless and often without common structure, not only between programs but also inside one and the same program.

It is not the intention to go through all of that here. Instead we give a general overview, and then provide an explicit example.

Generically, the usage of an event generator takes place in three steps.

1. Initialization step. It is here that all the basic characteristics of the coming generation are specified. This includes, among others, the following points.
   - To select which processes shall be generated.
   - To set constraints on the kinematics and the flavour characteristics of the process.
   - To define the physics scenario, e.g. masses of unknown particles.
   - To pick structure functions and other such aspects of the generation.
   - To switch off generator parts not needed, or in other ways to modify the coming generation.
   - To initialize the generator, e.g. to let it find maxima of differential cross-sections.
   - To book histograms, reset user counters, etc.
2. Generation loop. This is the main part of the run, and includes the following tasks.
   - To generate one event at a time.
   - To print a few of the events, to be able to check that everything is working as intended or to check up on anomalies.
   - To analyze events for the properties of interest.
   - To add the results of the analysis to histograms etc.
   - To write events to tape, and/or feed them to a detector program.
3. Finishing step. Here the tasks are as follows.
   - To print deduced cross-sections and other summary information from the generator.
   - To print histograms and other user-specified summary information.

To illustrate this structure, imagine a toy example, where one wants to simulate the production of a 300 GeV Higgs particle. In PYTHIA, a program for this might look something like the following.

```
C...Commonblocks.
      COMMON/LUJETS/N,K(4000,5),P(4000,5),V(4000,5)
      COMMON/LUDAT1/MSTU(200),PARU(200),MSTJ(200),PARJ(200)
      COMMON/LUDAT2/KCHG(500,3),PMAS(500,4),PARF(2000),VCKM(4,4)
      COMMON/LUDAT3/MDCY(500,3),MDME(2000,2),BRAT(2000),KFDP(2000,5)
      COMMON/PYSUBS/MSEL,MSUB(200),KFIN(2,-40:40),CKIN(200)
      COMMON/PYPARS/MSTP(200),PARP(200),MSTI(200),PARI(200)
      COMMON/PAWC/HBOOK(10000)

C...Number of events to generate; switch on proper processes.
      NEV=1000
      MSEL=0
      MSUB(102)=1
      MSUB(123)=1
      MSUB(124)=1

C...Select masses and kinematics cuts.
      PMAS(6,1)=140.
      PMAS(25,1)=300.
      CKIN(1)=290.
      CKIN(2)=310.

C...For simulation of hard process only: cut out unnecessary tasks.
```

```
      MSTP(61)=0
      MSTP(71)=0
      MSTP(81)=0
      MSTP(111)=0

C...Initialize and list partial widths.
      CALL PYINIT('CMS','p','p',16000.)
      CALL PYSTAT(2)

C...Book histograms.
      CALL HLIMIT(10000)
      CALL HBOOK1(1,'Higgs mass',50,275.,325.,0.)

C...Generate events; look at first few.
      DO 200 IEV=1,NEV
      CALL PYEVNT
      IF(IEV.LE.3) CALL LULIST(1)
      DO 100 I=1,N
  100 IF(K(I,2).EQ.25) HMASS=P(I,5)
      CALL HF1(1,HMASS,1.)
  200 CONTINUE

C...Print cross-sections
      CALL PYSTAT(1)
      CALL HISTDO

      END
```

Here 102, 123 and 124 are the three Higgs main Higgs production graphs $gg \to H$, $ZZ \to H$, and $WW \to H$, and MSUB(ISUB)=1 is the command to switch on process ISUB. Full freedom to combine subprocesses 'à la carte' is ensured by MSEL=0; ready-made 'menus' can be ordered with other MSEL numbers. The PMAS commands set the masses of the top quark and the Higgs itself, and the CKIN variables the desired mass range of the Higgs – a Higgs with a 300 GeV nominal mass actually has a fairly broad Breit-Wigner type mass distribution. The MSTP switches that come next are there to modify the generation procedure, in this case to switch off initial and final state radiation, multiple interactions among beam jets, and fragmentation, to give only the 'parton skeleton' of the hard process. The PYINIT call initializes PYTHIA, by finding maxima of cross-sections, recalculating the Higgs decay properties (which depend on the Higgs mass), etc. The decay properties can be listed with PYSTAT(2). Inside the event loop, PYEVNT is called to generate an event, and LULIST(1) to list it. The information used by LULIST(1) is the event record, stored in the commonblock LUJETS. Here one finds all particles produced, both final and intermediate ones, with information on particle species and event history (K array), particle momenta (P array) and production vertices (V array). In the loop over all particles produced, 1 through N, the Higgs particle is found by its code (K(I,2)=25, see below), and its mass is stored in P(I,5). Finally, PYSTAT(1) gives a summary of the number of events generated in the various allowed channels, and the inferred cross-sections.

In the run above, a typical event listing might look like the following.

```
                         Event listing (summary)

    I  particle/jet     KF    p_x      p_y      p_z       E          m

    1  !p+!           2212    0.000    0.000 8000.000 8000.000    0.938
```

```
    2  !p+!           2212    0.000    0.000-8000.000 8000.000    0.938
 ==============================================================================
    3  !g!              21   -0.505   -0.229   28.553   28.558    0.000
    4  !g!              21    0.224    0.041 -788.073  788.073    0.000
    5  !g!              21   -0.505   -0.229   28.553   28.558    0.000
    6  !g!              21    0.224    0.041 -788.073  788.073    0.000
    7  !H0!             25   -0.281   -0.188 -759.520  816.631  300.027
    8  !W+!             24  120.648   35.239 -397.843  424.829   80.023
    9  !W-!            -24 -120.929  -35.426 -361.677  391.801   82.579
   10  !e+!            -11   12.922   -4.760 -160.940  161.528    0.001
   11  !nu_e!           12  107.726   39.999 -236.903  263.302    0.000
   12  !s!               3  -62.423    7.195 -256.713  264.292    0.199
   13  !c~!             -4  -58.506  -42.621 -104.963  127.509    1.350
 ==============================================================================
   14  (H0)             25   -0.281   -0.188 -759.520  816.631  300.027
   15  (W+)             24  120.648   35.239 -397.843  424.829   80.023
   16  (W-)            -24 -120.929  -35.426 -361.677  391.801   82.579
   17  e+              -11   12.922   -4.760 -160.940  161.528    0.001
   18  nu_e             12  107.726   39.999 -236.903  263.302    0.000
   19  s         A       3  -62.423    7.195 -256.713  264.292    0.199
   20  c~        V      -4  -58.506  -42.621 -104.963  127.509    1.350
   21  ud_1      A    2103   -0.101    0.176 7971.328 7971.328    0.771
   22  d         V       1   -0.316    0.001  -87.390   87.390    0.010
   23  u         A       2    0.606    0.052   -0.751    0.967    0.006
   24  uu_1      V    2203    0.092   -0.042-7123.668 7123.668    0.771
 ==============================================================================
                  sum:  2.00     0.00     0.00     0.00 15999.98 15999.98
```

The event listing above is abnormally short, in part because some columns of informations were removed to make it fit into this text, in part because all initial and final state QCD radiation, all nontrivial beam jet structure, and all fragmentation was inhibited in the generation. Therefore only the skeleton of the process is visible. In line 1 and 2 one recognizes the two incoming protons. In lines 3 and 4 are incoming partons before initial state radiation and in 5 and 6 after – since there is no such radiation they coincide here. Line 7 shows the Higgs produced by gg fusion, 8 and 9 its decay products and 10–13 the second step decay products. Up to this point lines give a summary of the event history, indicated by the exclamation marks that surround particle names (and also reflected in the `K(I,1)` code, not shown). From line 14 onwards comes the particles actually produced in the final states, first in lines 14–16 particles that subsequently decayed, which have their names surrounded in brackets, and finally the particles and jets left in the end. Here this also includes a number of unfragmented jets, since fragmentation was inhibited. Ordinarily, the listing would have gone on for a few hundred more lines, with the particles produced in the fragmentation and their decay products. The final line gives total charge and momentum, as a convenient check that nothing unexpected happened. The first column of the listing is just a counter, the second gives the particle name and information on status and string drawing (the `A` and `V`), the third the particle flavour code (which is used to give the name), and the subsequent columns give the momentum components.

One of the main problems is to select kinematics efficiently. Imagine, e.g. that one is interested in the production of a single Z with a transverse momentum in excess of 50 GeV. If one tries to generate the inclusive sample of Z events, by the basic production graphs $q\overline{q} \to Z$, then most events will have low transverse momenta and will have to be discarded. That there are any events at all is due to the initial state generation machinery, which can build up transverse momenta for the incoming q and $\overline{q}$. However, the amount of initial state radiation can not be constrained beforehand. To increase the efficiency, one may therefore turn to the higher order processes $qg \to Zq$ and $q\overline{q} \to Zg$, where already the hard subprocess gives a transverse momentum to the Z. This transverse momentum can be constrained as one wishes, but again initial and final state radiation will smear the picture. If one were to set a $p_\perp$ cut at 50 GeV for the hard process generation, events would be missed where the Z was given only 40 GeV in the hard process but got the rest from initial state radiation. Not only therefore would cross-sections come out wrong, but so might the typical event shapes. In the end, it is therefore necessary to find some reasonable compromise, by starting the generation at 30 GeV, say, if one knows that only rarely do events below that fluctuate up to 50 GeV. Of course, most events will therefore not contain a Z above 50 GeV, and one will have to live with some inefficiency. It is not uncommon that only one event out of ten can be used, and occasionally it can be even worse.

If it is difficult to set kinematics, often it is easier to set the flavour content of a process. In a Higgs study, one might e.g. wish to study the decay $H^0 \to Z^0 Z^0$, with each $Z^0 \to e^+e^-$ or $\mu^+\mu^-$. It is therefore necessary to inhibit all other H^0 and Z^0 decay channels, and also to adjust cross-sections to take into account this change, all of which is fairly straightforward. However, if one instead wanted to consider the decay $Z^0 \to c\bar{c}$, with a D meson producing a lepton, not only would there then be the problem of different leptonic branching ratios for different D:s (which means that fragmentation and decay treatments would no longer decouple), but also that of additional $c\bar{c}$ pair production in parton shower evolution, to an extent that is unknown beforehand. In practice, it is therefore impossible to force D decay modes in a consistent manner.

4.2 Standardization

While each program is pretty much a world in itself, some standardization effort has been started. The first step was to agree on a standard code for particles and partons, so that everybody uses the same numbers, contrary to the previous chaotic situation. A detailed description of this standard can be found in [48]. Here we reproduce a few important examples, to give the flavour of the scheme.

1	d	11	e$^-$	21	g	111	π^0
2	u	12	ν_e	22	γ	211	π^+
3	s	13	μ^-	23	Z^0	311	K^0
4	c	14	ν_μ	24	W^+	321	K^+
5	b	15	τ^-	25	H^0	2112	n
6	t	16	ν_τ			2212	p

The numbers for mesons and baryons, codes above 100, are built up from the flavour content, using the quark codes, and from the spin of the particle. An antiparticle has minus the code of the particle.

A next step is to agree on a common format for the events generated, i.e. how particle momenta, particle species, and event history are to be stored. A set of commonblocks were proposed for this purpose in [49]. Although these have not been included directly into all generators, translation routines to the standard commonblocks are found for many of the programs. Not only does this simplify the analysis of events obtained with different generators, but to some extent it also allows the event generation chain to be shared between programs. For instance, if one program can be used to generate B mesons in some specific process, but is known to handle B decays poorly, the B decays could be inhibited there, and the undecayed B:s handed on to some other program.

The latest step in this evolution is found in [50], where also a standardization of particle decay data is proposed, among other things.

4.3 Program Limitations

Already in the previous section, we have considered some of the uncertainties in our current understanding of physics at the LHC. Many more examples could certainly have been found.

Another class of uncertainties comes from the presence of bugs, i.e. programming errors, in event generators. Given the complexity of LHC simulation, almost all programs have bugs. Some of these simply are typographical errors, others are correct transcriptions of incorrect formulae in the literature (e.g., the $WZ \to WZ$ matrix elements in PYTHIA were incorrect for several years because the published formulae were not correct), others are programs that work at current energies but break down when run in single precision at LHC energies, and yet others are real mistakes by the programmer. Given the size of these generators, an error can lie dormant for a long time before being discovered. Even when discovered, errors need not be correctly corrected by the authors. Indeed, in the recent LHC workshop we saw three such examples: the $gg \to \gamma\gamma$ matrix elements in PYTHIA, the $H \to \gamma\gamma$ partial width in ISAJET, and the $q\bar{q}, gg \to b\bar{b}$ matrix elements in HERWIG. In each of these cases, the first 'corrections' proposed by the authors did not solve the problem found by users, and repeated complaints were necessary to see some improvements in the situation. Errors that were more rapidly corrected are too numerous to be mentioned.

These examples do not imply a quality judgement on particular programs. Considering the size and complexity, there is no reason to say that event generators are any more error-prone than other comparable software. The message is rather that all critical studies should always be based on more than one event generator, and/or on analytical cross-checks of the generator results.

With the changing computer market, e.g. the emergence of RISC chips, one must keep in mind that programs may need to be modified for maximum efficiency, or to be run at all. SAVE statements did not used to be necessary, but are it today. In the long run, programs are also likely to leave Fortran 77 behind, and move on to Fortran 90, C++, or some other more powerful language. To give one exampe where Fortran 77 forces us to unnatural solutions, consider how the properties of the event record are split into integer- and real-type arrays. What one would like to have, and what more modern languages do offer, is the ability to define new derived data types (structures) with both integer and real components. Each particle would then be represented by a vector, with integer components for particle code and history, and real components for momenta and production vertices. The event record itself would become an array of such vectors.

5 Summary

In this paper we have given an introduction to and overview of the LHC event generators currently available. As behooves a report of this kind, emphasis has been put on the unknown aspects. In particular, we have stressed the need for several independent cross-checks of crucial results.

However, one can also take another point of view: considering the number of years left before actual turn-on, the quantity and quality of LHC/SSC event generators are probably far superior to those available for any other major new accelerator at a corresponding stage of planning. We today have standard methods for turning the crank on any basic process (also including new hypothetical particles), to include initial and final state radiation, beam jets, fragmentation, etc., and to arrive at fairly realistic representations of what LHC events might look like. If the details may be a bit uncertain, the general picture of events at the LHC is still fairly clear. As experience from the Tevatron, LEP and HERA finds its way into programs, the quality should improve further. Needless to say, much continued work by event generator authors is necessary, not just to improve on the expected, but also to prepare for the unexpected.

References

[1] R. Brun *et al.*, GEANT 3, CERN report DD/EE/84-1 (1989)

[2] 'Physics of the Superconducting Super Collider 1986', eds. R. Donaldson and J. Marx (1987)

[3] 'Experiments, Detectors, and Experimental Areas for the Supercollider', eds. R. Donaldson and M.G.D. Gilchriese (World Scientific, Singapore, 1988)

[4] 'Proceedings of the Summer Study on High Energy Physics in the 1990s', ed. S. Jensen (World Scientific, Singapore, 1989)

[5] 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990)

[6] T. Sjöstrand, Z. Physik **C42** (1989) 301

[7] F. Anselmo *et al.*, *in* 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990), Vol. II, p. 130

[8] F. James, Computer Physics Commun. **60** (1990) 329

[9] F. James, Rep. Prog. Phys. **43** (1980) 1145

[10] F.E. Paige and S.D. Protopopescu, *in* 'Physics of the Superconducting Super Collider 1986', eds. R. Donaldson and J. Marx (1987), p. 320

[11] H.-U. Bengtsson and T. Sjöstrand, Computer Physics Commun. **46** (1987) 43

[12] B.R. Webber and G. Marchesini, Nucl. Phys. **B310** (1988) 461;
G. Marchesini, B.R. Webber, G. Abbiendi, I.G. Knowles, M.H. Seymour and L. Stanco, Cavendish-HEP-90/26 (1991)

[13] R. Odorico, Computer Physics Commun. **32** (1984) 139; **59** (1990) 527

[14] K. Hahn and J. Ranft, Phys. Rev. **D41** (1990) 1463

[15] R.D. Field, G.C. Fox and R.L. Kelly, Phys. Lett. **119B** (1982) 439;
R.D. Field, Nucl. Phys. **B264** (1986) 687

[16] L. Angelini, L. Nitti, M. Pellicoro and G. Preparata, Phys. Rev. **D41** (1990) 2081

[17] B. Andersson, G. Gustafson and B. Nilsson-Almqvist, Nucl. Phys. **B281** (1987) 289

[18] I. Hinchliffe, personal communication

[19] A. Ali, B. van Eijk and I. ten Have, Nucl. Phys. **B292** (1987) 1

[20] H. Kuijf *et al.*, *in* 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990), Vol. II, p. 173

[21] F. Aversa, M. Greco, P. Chiappetta and J.P. Guillet, Z. Physik **C46** (1990) 253;
S.D. Ellis, Z. Kunszt and D.E. Soper, Phys. Rev. Lett. **64** (1990) 2121

[22] G. Ingelman and P.E. Schlein, Phys. Lett. **152B** (1985) 256

[23] P. Aurenche, R. Baier, M. Fontannaz and D. Schiff, Nucl. Phys. **B297** (1988) 661;
H. Baer, J. Ohnemus and J.F. Owens, Phys. Rev. **D42** (1990) 61

[24] F.A.Berends, W.T. Giele, H. Kuijf and B. Tausk, Nucl. Phys. **B357** (1991) 32

[25] See H. Kuijf *et al.*, *in* 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990), Vol. II, p. 91

[26] U. Baur and D. Zeppenfeld, Nucl. Phys. **B308** (1988) 127;
S. Willenbrock and D. Zeppenfeld, Phys. Rev. **D37** (1988) 1775

[27] U. Baur, E.W.N. Glover and J.J. van der Bij, Nucl. Phys. **B318** (1989) 106

[28] V. Barger, T. Han, J. Ohnemus and D. Zeppenfeld, Phys. Rev. **D41** (1990) 2782

[29] J.J. van der Bij and E.W.N. Glover, Nucl. Phys. **B321** (1989) 561

[30] B. van Eijk and R. Kleiss, *in* 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990), Vol. II, p. 183

[31] U. Baur and E.W.N. Glover, Nucl. Phys. **B347** (1990) 12

[32] U. Baur and E.W.N. Glover, Nucl. Phys. **B339** (1990) 38

[33] UA2 Collaboration, J. Alitti *et al.*, Phys. Lett. **B235** (1990) 363

[34] H. Baer, X. Tata and J. Woodside, Phys. Rev. **D42** (1990) 1450

[35] E. Eichten, I. Hinchliffe, K. Lane and C. Quigg, Rev. Mod. Phys. **56** (1984) 579; **58** (1985) 1065

[36] H. Plothow-Besch, *in* 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990), Vol. II, p. 146

[37] J.G. Morfín and W.-K. Tung, FERMILAB-PUB-90/74

[38] 'The Structure Function Package', H. Plothow-Besch, Proceedings of the 3rd Workshop on Detector and Event Simulation in High Energy Physics, Amsterdam, 8-12 April 1991

[39] A.H. Mueller, Phys. Lett. **104B** (1981) 161;
B.I. Ermolaev and V.S. Fadin, JETP Lett. **33** (1981) 269

[40] B.R. Webber, Ann. Rev. Nucl. Part. Sci. **36** (1986) 253

[41] Yu.L. Dokshitzer, V.A. Khoze and S.I. Troyan, *in* 'Perturbative QCD', ed. A.H. Mueller (World Scientific, Singapore, 1989), p. 241

[42] M. Jacob, CERN-TH.5821, to appear in the proceedings of the 25th International Conference on High Energy Physics (Singapore, 1990);
T. Hebbeker, talk at the Joint International Lepton-Photon Symposium & Europhysics Conference on High Energy Physics, Geneva, 25 July – 1 August 1991

[43] T. Sjöstrand, *in* 'Precision tests of the Standard Model at High Energy Colliders', eds. F. del Águila, A. Méndez and A. Ruiz (World Scientific, Singapore, 1991), p. 151

[44] T. Sjöstrand, Phys. Lett. **157B** (1985) 321

[45] M. Ciafaloni, Nucl. Phys. **B296** (1987) 249;
S. Catani, F. Fiorani and G. Marchesini, Nucl. Phys. **B336** (1990) 18;
G. Marchesini and B.R. Webber, Nucl. Phys. **B349** (1991) 617

[46] E.M. Levin and M.G. Ryskin, Phys. Rep. **189** (1990) 267

[47] T. Sjöstrand, Int. J. Mod. Phys. **A3** (1988) 751

[48] Particle Data Group, J. J. Hernández *et al.*, Phys. Lett. **B239** (1990) 1

[49] T. Sjöstrand *et al.*, *in* 'Z physics at LEP 1', eds. G. Altarelli, R. Kleiss and C. Verzegnassi, CERN 89-08, Vol. 3, p. 143

[50] F. Carminati *et al.*, *in* 'Large Hadron Collider Workshop', eds. G. Jarlskog and D. Rein, CERN 90-10 (1990), Vol. III, p. 52