

MINIGUIDE PYTHIA

Jean-Paul Guillaud Laboratoire d'Annecy de Physique des particules
Chemin de Bellevue BP 110 Annecy-le-Vieux CEDEX France

I) Introduction

II) General physics

- II-1) The hard processes and parton distributions
- II-2) The initial- and final-state radiation
- II-3) The matrix elements
- II-4) The parton showers
- II-5) The beam remnants
- II-6) Fragmentation and the Lund model
- II-7) The decays

III) Specific physics

- III-1) The light Standard Model Higgs production
- III-2) e^+e^- collisions
- III-3) hadron-hadron interactions
- III-4) J/ψ production

IV) Practical informations

- IV-1) PYTHIA printed information and the event listing
- IV-2) Particle codes list and particle decays table
- IV-3) Physics Processes numbering
- IV-4) kinematics cuts (CKIN)
- IV-5) Initializing PYTHIA and random sequences seed
- IV-6) Examples of pp physics generation
 - IV-6-1) $pp \rightarrow h^0 \rightarrow gg$ generator:
 - IV-6-2) $pp \rightarrow$ PYTHIA default min. bias and CMS recommended min. bias:
 - IV-6-3) $pp \rightarrow q\bar{q}$:
 - IV-6-4) $pp \rightarrow$ prompt photons:
 - IV-6-5) $pp \rightarrow Z^0$ or $pp \rightarrow Z^0 + jet$:
 - IV-6-6) $pp \rightarrow W^\pm$ or $pp \rightarrow W^\pm + jet$:
 - IV-6-7) $pp \rightarrow h^0 Z^0$ or $pp \rightarrow h^0 W^\pm$:
- IV-7) Frequently asked questions:
- IV-8) How to compile and load a program:
- IV-9) Downloading a tutorial script:

V) Acknowledgements

D) Introduction

The PYTHIA/JETSET package is composed of two programs that are merged into one single file. They provide a generation of high energy physics events, namely hard interactions between elementary particles such as electrons, positons, protons and antiprotons in various combinations. They contain theory and models for a number of physics aspects, including hard and soft interactions, parton distributions, initial- and final-state parton showers, multiple interactions, fragmentation and decay. They are largely based on original research (see Torbjörn Sjöstrand 1994 above-mentioned reference), but also they borrow many formulæ and other knowledge from the literature. They are thus limited by our current understanding of the underlying physics not covered by standard theory.

Although conceived separately, the PYTHIA and JETSET programs are today so often used together, and a border line between the two had become more and more artificial, that both programs are now maintained in common and refered under the common label PYTHIA.

JETSET was begun by members of the Lund theory since 1978 to generate e^+e^- physics (at PEP, PETRA then LEP), it can be used in the context of any hard process assuming «jet universality» (i.e. that the fragmentation is fundamentally the same for an e^+e^- or a pp event) and that the only differences are to be found in the parton-level processes involved.

PYTHIA has been added (about 1983) to generate the hadronic physics at very high energies (LEP,LHC), it generates the collisions between leptons, hadrons and gammas following the «QCD recipes». Because of the largeness of the strong coupling constant α_s , and because of the presence of the triple gluon vertex, QCD emission off quarks and gluons is especially prolific (parton showers), wherein a single initial parton may give rise to a whole bunch of partons in the final state. Also photon emission may give rise to sizeable effects.. The bulk of the bremsstrahlung corrections are universal, i.e. do not depend of the studied process, but only on one or a few key numbers, such as the momentum transfer scale of the process. Such universal corrections may be included to arbitrarily high orders, using a probabilistic language. Lastly, quarks and gluons are confined. In the program, a perturbative approach is used to describe the short distance interactions of quarks, leptons and gauge bosons. For leptons and colourless bosons, this approach is sufficient. However, for quarks and gluons it must be complemented with a picture for the hadronization process (which can be subdivided into fragmentation and decays), wherein the coloured partons are transformed into jets of colourless hadrons, photons and leptons. This process is still not yet understood from first principles, but has to be based on models. In one sense, hadronization effects are overwhelmingly large, since this is where the bulk of the multiplicity comes from. In another sense, the overall energy flow of a high energy event is mainly determined by the perturbative processes, with only a minor additional smearing caused by the hadronization step. All relevant variables are selected according to the desired probability distributions and thereby ensure randomness in the final events. The decays of fundamental resonances (Z^0, W^\pm, h^0, \dots) have branching ratios which are calculated dynamically (i. e. mass dependent) they give either leptons, quarks ($W \rightarrow qq'$) or resonances. Then matrix elements (spin correlation) when available are taken in account, if not decays are isotropic. In the program, the generation of events as detailed as they could be observed by a «perfect» detector is not done in one step but

rather by factorizing the full problem into a number of components, each of which being reasonably handled accurately. Basically, this means that the hard process is used as input to generate bremsstrahlung corrections and that the result of it is thereafter left to hadronize. If the full problem is too complicated to be solved in one go, one tries to subdivide it into smaller tasks of manageable proportions (i.e. branching of one object into two, or at least into a very small number, each of which being free to branch in its turn). A lot of bookkeeping is involved but much is of repetitive nature..

The logic of the package (as long as PYTHIA and JETSET might be delimited) may be divided in 3 steps:

- 1 define the process(es) you want to study (all events will then be generated according to this specification),
- 2 generate (with PYTHIA) the showers of **the initial states** and the beam fragments,
- 3 generate (with JETSET) the showers of **the final states** the fragmentation “à la Lund” (confirmed by data from PETRA and PEP) and the decays.

The program is very versatile but the price to pay for this is having a large number of adjustable parameters and switches for alternative modes of operation BUT **all switches and parameters** are provided with **sensible default values** based on our best understanding of the physics involved. A new user can therefore disregard all the fancy options. If anything goes wrong that the program can catch, an error message will be printed and the treatment of the corresponding event will be cut short. In serious cases, the program will abort. If no error appears it is OK. For details on the hard process, initial- and final-state radiation, fragmentation, decays and the Lund model see **chapter II**. For details concerning more specific physics (my personal choice..) and few examples (namely a $h^0 \rightarrow gg$ generator) see **chapter III. Practical informations** (mainly those I needed for building a sensible **pp physics** generator) and frequently asked questions (at least mine) and answers **are given in chapter IV** (with the possibility to download a script tutorial interface to PYTHIA).

II) General physics

In this section, an overview of the main physics features of JETSET and PYTHIA will be given.

For the description of a typical high-energy event, the collision may be described (in some semblance of a time order) following 7 steps:

- 1 Initially, the two beams are coming in towards each other. Normally, each particle is characterized by a set of parton distributions functions, which defines the partonic substructure in terms of flavour composition and energy sharing.
- 2 One shower initiator parton from each beam starts off a sequence of branchings such as $q \rightarrow qg$, which build up an initial-state shower.
- 3 One incoming parton from each of the two showers enters the hard process, where then a number of outgoing partons are produced, usually two. It is the nature of this process that determines the main characteristics of the event.
- 4 Also the outgoing partons may branch (multiple interactions), to build up final-state showers.
- 5 When a shower initiator is taken out of a beam particle, a beam remnant is left behind. This remnant may have an internal structure, and a net colour charge that relates it to the rest of the final state.
- 6 The QCD confinement mechanism ensures that the outgoing quarks and gluons are not observable, but instead fragment to colour neutral hadrons.
- 7 Many of the produced hadrons are instable and decay further.

Conventionally, only quarks and gluons are counted as partons, while leptons and photons are not. If pushed *ad absurdum* this may lead to some unwieldy terminology.

Therefore, where it does not matter, an electron or a photon will be considered in the «partonic» substructure of an electron, branchings $e \rightarrow e\mathbf{g}$ will be lumped together with other parton shower branchings such as $q \rightarrow qg$ and so on. With this notation, the division into the above seven points applies equally well to an interaction between two leptons, between a lepton and a hadron, and between two hadrons.

In the following subsections, the above seven points will be surveyed, not in the same order as given here, but rather in which they appear in the program execution, i. e. starting with the hard process.

II-1) The hard processes and parton distributions

In JETSET, the main hard process is $e^+e^- \rightarrow \mathbf{g}^* / Z^0 \rightarrow q\bar{q}$ (the * is used to denote that the photon must be off the mass shell). The quark q in the final state may be u, d, s, c, b or t (the flavour in each event being picked up at random, according to the relative couplings, evaluated at the hadronic c.m. energy). Also there is processes with the ggg and ggg final states, as expected in onium 1^{--} decays such as Υ but toponium will decay weakly much too fast to be of any interest.. PYTHIA contains a very rich selection of

processes. These may be classified according to the physics scenario, i. e. that one may distinguish 14 major groups:

- * Hard QCD processes (e. g. $qg \rightarrow qg$).
- * Soft QCD processes, such as diffractive and elastic scattering and minimum bias events.
- * Heavy flavour production (e. g. $gg \rightarrow t\bar{t}$).
- * Prompt-photon production (e. g. $qg \rightarrow qg$).
- * Photon-induced processes (e. g. $gg \rightarrow q\bar{q}$).
- * Deep inelastic scattering (e. g. $ql \rightarrow ql$).
- * W/Z production, such as the $e^+e^- \rightarrow g^* / Z^0$ of JETSET or $q\bar{q} \rightarrow W^+W^-$.
- * Standard model Higgs production, where the Higgs is reasonably light and narrow, and can therefore still be considered as a resonance.
- * Gauge boson scattering processes, such as $WW \rightarrow WW$, when the Standard Model Higgs is so heavy and broad that resonant and non-resonant contributions have to be considered together.
- * Non-standard Higgs particle production, within the framework of a two-Higgs doublet scenario with three neutral (h^0 , H^0 and A^0) and two charged (H^\pm) Higgs states.
- * Production of new gauge bosons as a Z' .
- * Production of fourth-generation fermions.
- * Leptoquark (L_Q) production.
- * Deviations from Standard Model processes, e. g. due to contact interactions or a strongly interacting gauge boson sector. These do not always appear as separate processes, but may just be options to some of the processes above.

The cross section for a given process $ij \rightarrow k$ is given by

$$\mathcal{S}_{ij \rightarrow k} = \int dx_1 \int dx_2 f_i^1(x_1) f_j^2(x_2) \hat{\mathcal{S}}_{ij \rightarrow k}$$

where $\hat{\mathcal{S}}_{ij \rightarrow k}$ is the cross section for the hard partonic process, as codified in the matrix elements for each specific process. For processes with many particles in the final state it is replaced by an integral over the allowed final-state phase space. The $f_i^a(x)$ are the parton-distribution functions, which describe the probability to find a parton i inside beam particle a , with parton i carrying a fraction x of the total a momentum. Actually, parton distributions also depend on some momentum scale Q^2 that characterizes the hard process.

Parton distributions are most familiar for hadrons, such as the proton. Hadrons are inherently composite objects, made up of quarks and gluons. Since a derivation of hadron parton distribution from first principles does not yet exist, the program necessary relies on parametrizations, where experimental data are used in conjunction with the evolution equations for the Q^2 dependence.

For a fundamental particle, such as the electron, the program introduces parton distributions (the function $f_e^a(x)$ parametrizes the probability that the electron that takes

part in the hard process retains a fraction x of the original energy, the rest being radiated (into photons) in the initial state. Note that, contrary to the proton case, electron parton distributions are calculable from first principles and reduce to the d function above for $Q^2 \rightarrow 0$.

The electron may also contain photons, and the photon may in its turn contain quarks and gluons. The internal structure of the photon contains a point-like part, which is perturbatively calculable, and a vector dominance part, which is not. In the program, the photon parton distributions are parametrized just as the hadrons ones. Since the electron contains quarks and gluons, hard QCD processes like $qg \rightarrow qg$ appear in pp , ep and e^+e^- collisions (this possibility of reusing one and the same hard process in different contexts justifies *a posteriori* the parton distribution approach).

The last generalization to point out concerns the processes $e^+e^- \rightarrow g^*/Z^0$ (studied in e^+e^- colliders) and $q\bar{q} \rightarrow g^*/Z^0$ (studied in hadrons colliders) i. e. more specifically $f\bar{f} \rightarrow g^*/Z^0$ where f denotes any fundamental fermion (quark, lepton, neutrino etc.). In a hadron collider, the interesting cross section is thus a sum over several different initial states (e. g. $u\bar{u} \rightarrow g^*/Z^0$ and $d\bar{d} \rightarrow g^*/Z^0$).

II-2) The initial- and final-state radiation

In every process that contains coloured and/or charged objects in the initial or final state, gluon and/or photon radiation may give large corrections to the overall topology of events. Starting from a basic $2 \rightarrow 2$ process, this kind of corrections will generate $2 \rightarrow 3$, $2 \rightarrow 4$ and so on, final state topologies and when the available energy increases this hard emission becomes more and more important relative to fragmentation. The traditional approach to handle perturbative corrections (the matrix –element method i. e. to calculate Feynman diagrams order by order taking in account exact kinematics, the full interference and the helicity structure) becomes increasingly difficult in higher orders with the limitation of becoming less relevant with the increase of the available energy that increases the phase space available for gluon emission. It could be replaced by the emission of multiple soft gluons and we have indirect but strong evidence that it plays a significant role in building up the event structure. However, this perturbative expansion behaves better at higher energies owing to the running of α_s . It could also be replaced by the «parton shower method», using not the full matrix-element expressions but approximations derived by simplifying the kinematics, and the interference and the helicity structure. One could expect that this method gives a good description of the substructure of jets with a limitation on the predictive power for the rate of well separated jets (i. e. the 2/3/4/5-jets composition) and on absolute α_s determinations. These two approaches are complementary in many respects. Because of its simplicity and its flexibility, the second one (the parton shower option) is used as a first choice while the first one (the matrix-element method) should be used for absolute α_s determinations and other specialized studies.

II-3) The matrix elements

Matrix elements are especially made use of in the JETSET implementation of the process $e^+e^- \rightarrow g^*/Z^0 \rightarrow q\bar{q}$.

For initial-state QED radiation, a first order (unexponentiated) description is adopted. Events are subdivided in two classes: those where a photon is radiated above some minimum energy and those without such a photon (in this class, the soft and virtual corrections have been lumped together to give a total event rate that is correct up to one loop). For final-state QCD radiation, the default is the «parton shower method» but the matrix-elements options are also frequently used. In the definition of 3- or 4-jet events, a cut is introduced whereby it is required that any two partons have an invariant mass bigger than some fraction of the c.m. energy. 3-jet events which do not fulfil this requirement are lumped with the 2-jet ones. The first-order matrix-element option which only contains 3- and 2-jet events therefore involves no ambiguity. In second order, where also 4-jets have to be considered, a main issue is what to do with 4-jet events that fail the cuts. Depending on the choice of recombination scheme, whereby the two nearby partons are joined into one, different 3-jet events are produced. Therefore the second-order differential 3-jet rate has been the subject of some controversy and JETSET actually contains two different implementations.

By contrast PYTHIA does not contain any full higher-order matrix elements, with loop contributions included. There are a few cases where higher-order matrix elements are included at the Born level e.g. the case of W production at a hadron collider, which is contained in the lowest-order process $q\bar{q}' \rightarrow W$. PYTHIA also contains the two first-order processes $qg \rightarrow Wq'$ and $q\bar{q}' \rightarrow Wg$. The cross section for these processes are divergent when $P_T \rightarrow 0$. In this region a correct treatment would therefore have to take in account loop corrections, which are not available in PYTHIA. Depending on the physics application, one could then use PYTHIA in one of two ways: In the region of small P_T , the preferred option is lowest-order matrix elements combined with parton showers. For the production of a W at large P_T , the shower approach has been shown to be appropriate (Gabriela Miu and Torbjörn Sjöstrand Phys. Lett. B449 (1999)313) but for other processes the shower approach is too imprecise to give the right cross section; Here it is advantageous to generate first-order events and then add showers only to describe additional softer radiation.

II-4) The parton showers

The separation of radiation into initial- and final-state showers is arbitrary, but very convenient. Similarly, the distinction of emission as coming from the quark or from the antiquark is arbitrary. In general, the assignment of a radiation to a given «mother parton» is a good approximation for an emission close to the direction of motion of that parton, but not for a wide angle emission in between two jets, where interference terms are expected to be important.

In both initial- and final-state showers, the structure is given in terms of branchings $a \rightarrow bc$, specifically $e \rightarrow e\mathbf{g}$, $q \rightarrow q\mathbf{g}$, $q \rightarrow q\mathbf{g}$, $g \rightarrow g\mathbf{g}$ and $g \rightarrow q\bar{q}$. Each of these processes is characterized by a splitting kernel $P_{a \rightarrow bc}(z)$. The branching rate is proportional to the integral $\int P_{a \rightarrow bc}(z) dz$. The z value picked for a branching describes the energy sharing with daughter b taking a fraction z and daughter c the remaining $1-z$ of the a energy. Once formed, the daughters b and c may in turn branch, and so on.

Each parton is characterized by some «virtuality scale» Q^2 , which gives an approximate sense of time ordering to the cascade. In the initial-state shower, Q^2 values are gradually

increasing as the hard scattering is approached, while Q^2 is decreasing in the final-state showers. Shower evolution is cut off at some lower scale Q_0 , typically around 1 GeV for QCD branchings. The same cut-off scale is also used to regularize the soft gluon emission divergences in the splitting kernels.

From above, a maximum scale Q_{\max} is introduced, where the showers are matched to the hard interaction itself. The relation between Q_{\max} and the kinematics of the hard scattering is uncertain, and the choice made can strongly affect the amount of well separated jets. Despite a number of common traits the initial- and final-state radiation machineries are in fact quite different. For historical reasons, the final-state shower is found in JETSET and the initial-state one in PYTHIA.

Final-state showers are time-like (i. e. partons have $m^2 = E^2 - p^2 \geq 0$). The evolution variable Q^2 of the cascade is therefore in JETSET associated with the m^2 of the branching parton. Starting from Q_{\max}^2 , an original parton is devolved downwards in Q^2 until a branching occurs. The selected Q^2 value defines the mass of the branching parton, and the z of the splitting kernel the parton energy division between its daughters. These daughters, in turn, may now evolve downwards (with maximum virtuality defined by kinematics) down to the Q_0 cut-off.

In contrast to final-state showers, initial-one are space-like, i. e. in the sequence of branchings $a \rightarrow bc$ that lead up from the shower initiator to the hard interaction, particles a and b have $m^2 = E^2 - p^2 < 0$. The «side branch» particle c , which does not participate in the hard scattering, may be on the mass shell or have a time-like virtuality. Such a time-like shower will evolve off it, rather like the final-state radiation here above. To first approximation, the evolution of the space-like main branch is characterized by the evolution variable $Q^2 = -m^2$, which is required to be strictly increasing along the shower, i. e. $Q_b^2 > Q_a^2$. Corrections to this picture are basically absent in PYTHIA (but some options exist).

Initial-state radiation is handled within the backwards evolution scheme. In this approach, the choice of the hard scattering is based on the use of evolved parton distributions, which means that the inclusive effects are already included. The construction of the exclusive showers is done starting from the two incoming partons at the hard interaction, tracing the showers «backwards in time» back to the two shower initiators. In other words, given a parton b , one tries to find a parton a that branched into b . The evolution is therefore in terms of a sequence of decreasing space-like virtualities Q^2 and increasing momentum fractions x . Branching on the two sides are interleaved in a common sequence of decreasing Q^2 values. In this formalism, there is no real distinction between gluon and photon emission although some of the details actually do differ..

The initial- and final-state radiation shifts around the kinematics of the original hard interaction. In deep inelastic scattering, this means that the x and Q^2 values that can be derived from the momentum of the scattered lepton do not agree with the values originally picked. In high- P_T processes, it means that one no longer has two jets with opposite and compensating P_T , but more complicated topologies. Effects of any original

kinematics selection cuts are therefore smeared out, an unfortunate side-effect of the parton-shower approach.

II-5) The beam remnants

In a hadron-hadron collision, the initial-state radiation algorithm reconstructs one shower initiator in each beam. This initiator only takes some fraction of the total beam energy, leaving behind a beam remnant that takes the rest. For a proton beam, a u quark initiator would leave behind a ud diquark beam remnant with an antitriplet colour charge. The remnant is therefore colour-connected to the hard interaction, and forms part of the same fragmenting system. To take in account the motion of quarks inside the original hadron (as required by the uncertainty principle), a primordial transverse momentum is assigned to the shower initiator according to some suitable distribution and the recoil is taken up by the beam remnant. Often the remnant is more complicated, e. g. a g initiator would leave behind a uud proton remnant system in a colour octet state which can be subdivided into a colour triplet quark and a colour antitriplet diquark, each of which being colour-connected to the hard interaction. The energy sharing between these two remnant objects, and their relative transverse momentum, introduces additional degrees of freedom, which are not understood from first principles..

Naïvely, one would expect an ep event to have only one beam remnant, and an e^+e^- event none. This is not always correct, e. g. a $gg \rightarrow q\bar{q}$ interaction in an e^+e^- event would leave behind the e^+ and e^- as beam remnants, and a $q\bar{q} \rightarrow gg$ interaction in resolved photoproduction in an e^+e^- event would leave behind one e^\pm and one q/\bar{q} in each remnant. Corresponding complications occur for photoproduction in ep events.

In hadron-hadron collisions, each of the beam particles contains a multitude of partons, and so the probability for several interactions in one and the same event need not be negligible. In principle these additional interactions could arise because one single parton from one beam scatters against several different partons from the other beam, or because several partons from each beam take place in separate $2 \rightarrow 2$ scatterings. Both are expected, but combinatorics should favour the latter, which is the mechanism considered in PYTHIA.

The dominant $2 \rightarrow 2$ QCD cross sections are divergent for $P_T \rightarrow 0$, and drop rapidly at large P_T . Probably the lowest-order perturbative cross sections will be regularized at small P_T by colour coherence effects: an exchanged gluon of small P_T has a large transverse wave function and can therefore not fully resolve the individual colour charges of the two incoming hadrons; it will only couple to an average colour charge that vanishes in the limit $P_T \rightarrow 0$. In the program, some effective $P_{T\min}$ scale is therefore introduced, below which the perturbative cross section is either assumed completely vanishing or at least strongly damped. Phenomenologically, $P_{T\min}$ comes out to be a number of the order of 1.5-2.0 GeV.

In a typical minimum bias event one therefore expects to find one or a few scatterings at scales around or a bit above $P_{T\min}$, while a high- P_T event also may have additional scatterings at the $P_{T\min}$ scale. The probability to have several high- P_T scatterings in the same event is small, since the cross section drops so rapidly with P_T . Arguments can be given why $P_{T\min}$ should be energy-dependent, and currently a slow increase with energy is assumed in the program.

In conclusion, the understanding of multiple interaction is still primitive. PYTHIA therefore contains several options, with a fairly simple one as default.

II-6) Fragmentation and the Lund model

QCD perturbation theory (in terms of quarks and gluons) is valid at short distances. At long distances, QCD becomes strongly interacting and perturbation theory breaks down. In this confinement regime, the coloured partons are transformed into colourless hadrons, a process called fragmentation (or hadronization). In lack of understanding the fragmentation from first principles (i. e. starting from the QCD Lagrangian), the program uses a phenomenological model (among the many existing) of string fragmentation, in the form of the time-honoured «Lund model» but the independent fragmentation (IF) also exist, for those who wish to study the importance of string effects.

The fragmentation process as a whole is described in terms of one or a few underlying branchings, of the type string \rightarrow hadron + remainder string and so on. At each branching, probabilistic rules are given for the production of new flavours, for the sharing of energy and momentum between the products. To understand fragmentation models, it is useful to start with the simplest possible system, a colour-singlet $q\bar{q}$ 2-jet event (as produced in a e^+e^- annihilation). Lattice QCD studies lend support to a linear confinement picture (in the absence of dynamical quarks): the energy stored in the colour dipole field between a charge and an anticharge increases linearly with the separation between the charges (if the short distance Coulomb term is neglected). This is quite different from the behaviour in QED, and is related to the presence of a triple gluon vertex in QCD. The details are not yet well understood however.

The assumption of linear confinement provides the starting point for the string model. As the q and the \bar{q} partons move apart (from their common production vertex), the physical picture is that of a colour flux tube being stretched between the q and the \bar{q} . The transverse dimensions of the tube are of typical hadronic sizes, roughly 1 fm. If the tube is assumed to be uniform along its length, this automatically leads to a confinement picture with a linearly rising potential. In order to obtain a Lorentz covariant and causal description of the energy flow due to this linear confinement, the model uses the dynamics of the massless relativistic string with no transverse degrees of freedom. The one-dimensional string can be thought of as the axis of a cylindrically symmetric flux tube. From hadron spectroscopy, the string constant (the amount of energy per unit length) is deduced to be $K=1$ GeV/fm (K effectively corresponds to a «mass density» along the string, therefore the expression «massless» is somewhat of a misnomer). As the q and the \bar{q} move apart, the potential energy stored in the string increases and the string may break by the production of a new $q'\bar{q}'$ pair, so that the system splits into two colour-singlet systems $q\bar{q}'$ and $q'\bar{q}$. If the invariant mass of either of these string pieces is large enough, further breaks may occur. In the Lund string model, the string break-up process proceeds until only on-mass-shell hadrons remain, each hadron corresponding to a small piece of string with a quark in one end and an antiquark in the other.

In order to generate the quark-antiquark $q'\bar{q}'$ pair the Lund model invokes the idea of quantum mechanical tunnelling that implies a suppression of heavy-quark production (u:d:s:c = 1:1:0.3:10⁻¹¹) and that leads to a flavour-independent Gaussian spectrum for the P_T of the $q'\bar{q}'$ pairs. This P_T is locally compensated (between the quark and the

antiquark of the pair) since the string is assumed to have no transverse excitations. Due to the previous ratios, charm and heavier quarks are not expected to be produced in the soft fragmentation but only in perturbative parton-shower branchings $g \rightarrow q\bar{q}$. When the quark and antiquark from two adjacent string breakings are combined to form a meson, in lack of prediction of the string model, the choice between pseudoscalar and vector mesons is with the ratio 1 to 3 from counting the number of spin states, multiplied by some wave-function normalization factor, which should disfavour the heavier states. A tunnelling mechanism can also be used to produce baryons (this is still a poorly understood area): In the simplest possible approach, a diquark in a colour antitriplet state is just treated like an ordinary antiquark, such that a string can break either by quark-antiquark or antidiquark-diquark pair production.

In general the different string breaks are causally disconnected (the breaks can be described in any convenient order, e. g. from the quark end inwards), so that an iterative scheme can be used for the fragmentation: Assume an initial quark q moving out along the $+z$ axis with the antiquark going out in the opposite direction. By the production of a $q_1\bar{q}_1$ pair, a meson $q_1\bar{q}_1$ is produced, leaving behind an unpaired quark q_1 . A second pair $q_2\bar{q}_2$ may now be produced, to give a new meson $q_1\bar{q}_2$ etc. At each step, the produced hadron takes some fraction of the available energy and momentum. This process may be iterated until all energy is used up, with some modifications close to the \bar{q} end of the string in order to make total energy and momentum come out right. The choice of starting from the quark end is arbitrary, however a process described in terms of starting at the \bar{q} end of the system, fragmenting towards the q end should be equivalent. This «left-right» symmetry constrains the allowed shape of the fragmentation function $f(z)$ where z is the fraction of the remaining light cone momentum $E \pm p_z$ (+ for the q jet, - for the \bar{q} one) taken by each new particle. The resulting «Lund symmetric fragmentation function» has two free parameters, which are determined from data.

If several partons are moving apart from a common origin, the details of the string drawing become more complicated. For a $q\bar{q}g$ event, a string is stretched from the q end via the g to the \bar{q} end, i.e. the gluon is a kink on the string, carrying energy and momentum. As a consequence, the gluon has two string pieces attached, and the ratio of gluon to quark string force is 2, a number which can be compared with the ratio of colour charge Casimir operators, $N_C / C_F = 2 / (1 - 1 / N_C^2) = 9 / 4$. Note that the factor 2 above does not depend on the kinematical configuration: a smaller opening angle between two partons corresponds to a smaller string length drawn out per unit time, but also to an increased transverse velocity of the string piece, which gives an exactly compensating boost factor in the energy density per unit string length. The $q\bar{q}g$ string will fragment along its length: This means that there is one fragmenting string piece between q and g and a second one between g and \bar{q} . One hadron is «straddling» both string pieces, i. e. sitting around the gluon corner. The rest of the particles are produced as in two simple $q\bar{q}$ strings, but strings boosted with respect to the overall c.m. frame. For events that involve many partons, there may be several possible topologies for their ordering along the string. An example would be a $q\bar{q}g_1g_2$ event (the gluon indices label different gluon momentum vectors), where the string can connect the partons in either of the sequences $q - g_1 - g_2 - \bar{q}$ and $q - g_2 - g_1 - \bar{q}$. The matrix elements that are calculable in perturbation theory contain interference terms between these two possibilities, which

means that the colour flow is not always well-defined. Fortunately the interference terms are down in magnitude by a factor $1/N_C^2$, where $N_C=3$ (the number of colours), so approximate recipes can be found. In the leading log shower description, on the other hand, the rules for the colour flow are well-defined..

Moreover soft gluon exchanges between partons are tacitly assumed to not mess up the original colour assignment (this is likely the case but has not been proven).. Alternative «colour rearrangement» scenarios are available for $e^+e^- \rightarrow W^+W^- \rightarrow q_1\bar{q}_2q_3\bar{q}_4$.

II-7) The decays

A large fraction of the particles produced by fragmentation are unstable and subsequently decay into the observable stable (or almost stable) ones. It is therefore important to include all particles with their proper mass distributions and decay properties, This is less trivial than it may sound: While a lot of experimental information is available, there is also very much that is missing.

Normally it is assumed that decay products are distributed according to phase space, i. e. that there is no dynamics involved in their relative distribution. However, in many cases additional assumptions are necessary, e. g. for semileptonic decays of charm and bottom hadrons one needs to include the proper weak matrix elements. Particles may also be produced polarized and impart a non-isotropic distribution to their decay products. Many of these effects are not at all treated by the program. In fact, spin information is not at all carried along but has to be reconstructed explicitly when needed.

The normal decay treatment is handled by JETSET, making use of a set of tables where branching ratios and decay modes are stored. In PYTHIA a separate decay treatment exists, used exclusively for a specific list of particles:

$Z^0, W^\pm, h^0, Z^0, W^\pm, H^0, A^0, H^\pm, H_{tech}^0, R^0, q^*, l^*$, and the leptoquark L_Q . These particles are hereunder called «resonances», and contrast the «particle decay» treatment of JETSET with the «resonance decay» one of PYTHIA. Of course, this is just a matter of terminology: a particle like the r could also be called a resonance. What characterizes a PYTHIA resonance is that partial widths and branching ratios are calculated dynamically, as a function of the actual mass of the particle. Therefore not only do branching ratios change between an h^0 of nominal mass 100 GeV and one of 200 GeV, but also for a Higgs of nominal mass 200 GeV, the branching ratios would change between an actual mass of 190 GeV and 210 GeV, say. This is particularly relevant for reasonably broad resonances, and in threshold regions. For an approach like this to work, it is clearly necessary to have perturbative expressions available for all partial widths, which is one reason why a corresponding treatment would not be the same for an ordinary hadronic resonance like the r (**note that H^0 and H^0 of PYTHIA 5.7 are now h^0 and H^0 in PYTHIA 6.1.**

The decay products of PYTHIA resonances are typically quarks, leptons, or other resonances, e. g. $W \rightarrow q\bar{q}$ or $h^0 \rightarrow W^+W^-$. In decays to quarks, parton showers are automatically added to give a more realistic multijet structure, and one may also allow photon emission off leptons. If the decay products in turn are resonances, further decays are necessary. Often spin information is available in resonance decay matrix elements,

contrary to the normal state of affairs in ordinary particle decays. This means that the angular orientations in the two decays of a W^+W^- pair are properly correlated. Occasionally, the information is not available and then resonances decay isotropically.

The top is handled like a resonance, too short-lived to hadronize before it decays.

III) Specific physics

This chapter is restricted to few points of interest (my personal choice to shorten this miniguide). For specific other needs, reader should consult the 1994 reference given in this miniguide header.

III-1) The light Standard Model Higgs production

In a hadron collider, the main processes are gluon fusion ($gg \rightarrow h^0$), $Z^0 Z^0$ fusion ($f_i \bar{f}_i \rightarrow f_j \bar{f}_j h^0$) and $W^+ W^-$ fusion ($f_i \bar{f}_i \rightarrow f_k \bar{f}_k h^0$) (i. e. processes 102, 123 and 124). To generate such a Higgs the FORTRAN code should be:

```
call PYGIVE('msel = 0')           ! requires full user control.
call PYGIVE('msub(102) = 1')      ! gluon fusion.
call PYGIVE('msub(123) = 1')      ! Z0Z0 fusion.
call PYGIVE('msub(124) = 1')      ! W+W- fusion.
```

This piece of code should be inserted ahead of the PYTHIA initialization (see **IV-5 a**).

Note: the call to PYGIVE (**LUGIVE in PYTHIA 5.7**) wrapper on the common block changes is not necessary, but provides some extra checks and is therefore safer.

In the latter two processes, the emission of the space-like W/Z bosons off quarks are taken in account. A process like $gg \rightarrow gh^0$ (i. e. process 113) with a Higgs recoiling against a gluon jet is effectively generated by initial-state corrections to process 102 (i. e. $gg \rightarrow h^0$), thus in order to avoid double counting these processes should not be switched «on» simultaneously. Process 102 should be used for inclusive production of Higgs and 111 (i. e. $f_i \bar{f}_i \rightarrow gh^0$) 112 (i. e. $f_i g \rightarrow f_i h^0$) 113 (i. e. $gg \rightarrow gh^0$) for the study of the Higgs subsample with high transverse momentum.

Warning: Processes 111-113 contain very lengthy and potentially numerically unstable loop integrals, so should be used with caution.

In $e^+ e^-$ annihilation, associated production of an h^0 with a Z^0 (i. e. process 24 $f_i \bar{f}_i \rightarrow Z^0 h^0$) is usually the dominant one close to the threshold, while the $Z^0 Z^0$ fusion and $W^+ W^-$ fusion (i. e. processes 123 and 124) win out at high energies. gg fusion (i. e. process 103 $gg \rightarrow h^0$) may also be of interest, in particular when the possibilities of bremsstrahlung photons and backscattered photons are included. Process 110 (i. e. $f_i \bar{f}_i \rightarrow gh^0$) which gives an h^0 in association with a g , is a loop process and is therefore suppressed in rate. The branching ratios of the Higgs are very strongly dependent on the mass. The program calculates them as a function of the actual Higgs mass, i. e. not just at the nominal mass.

Since the Higgs is a spin-0 particle, it decays isotropically. In decay processes such as $h^0 \rightarrow W^+ W^- \rightarrow 4$ fermions angular correlations are included. Also in processes 24 (i. e. $f_i \bar{f}_i \rightarrow Z^0 h^0$) and 26 (i. e. $f_i \bar{f}_i \rightarrow W^\pm h^0$), Z^0 and W^\pm decay angular distributions are correctly taken in account.

III-2) e^+e^- collisions

The main annihilation process is $e^+e^- \rightarrow Z^0$ (i. e. process 1) where in fact the full \mathbf{g}^*/Z^0 interference structure is included. This process can be used, with some confidence, for c.m. energies from about 4 GeV upwards. This is the default process obtained with **MSEL=1**, i.e. when you do not change anything yourself.

III-3) hadron-hadron interactions

In hadron-hadron interactions, the total hadronic cross section for $AB \rightarrow \text{anything}$, \mathbf{s}_{tot}^{AB} , is calculated using the parametrization of Donnachie and Landshoff (Phys. Lett. **B296** (1992)227.): it appears as the sum of two terms (one Pomeron and one Reggeon):

$$\mathbf{s}_{tot}^{AB}(s) = X^{AB} s^e + Y^{AB} s^{-h},$$

where $s = E_{cm}^2$. The powers $e = 0.0808$ and $h = 0.4525$ are expected to be universal, whereas the coefficients X^{AB} and Y^{AB} are specific to each initial state with the Pomeron term X^{AB} being expected to be the same at high energies for the particle and antiparticle (i. e. $X^{AB} = X^{\bar{A}\bar{B}}$). Parametrizations not provided by Donnachie and Landshoff have been calculated in the same spirit, making use of quark counting rules (G.A. Schuler and T. Sjöstrand, Nucl. Phys. **B407** (1993)539.):

$$\mathbf{s}_{tot}^{AB}(s) = \mathbf{s}_{el}^{AB}(s) + \mathbf{s}_{sd(XB)}^{AB}(s) + \mathbf{s}_{sd(AX)}^{AB}(s) + \mathbf{s}_{dd}^{AB}(s) + \mathbf{s}_{nd}^{AB}(s).$$

Here «el» is the elastic process $AB \rightarrow AB$, «sd(XB)» the single diffractive $AB \rightarrow XB$, «sd(AX)» the single diffractive $AB \rightarrow AX$, «dd» the double diffractive $AB \rightarrow X_1 X_2$ and «nd» the non-diffractive ones. The non-diffractive component is identified with the «minimum bias» physics (a practical but not unambiguous choice). Its cross section is given by «whatever is left».

At not too large squared momentum transfers t , the elastic cross section can be approximated by a simple exponential fall-off. The optical theorem gives (neglecting the small real part): $\frac{d\mathbf{s}_{el}}{dt} = \frac{\mathbf{s}_{tot}^2}{16\mathbf{p}} \exp(B_{el}t)$, and $\mathbf{s}_{el} = \mathbf{s}_{tot}^2 / 16\mathbf{p}B_{el}$. The elastic slope parameter is parametrized by $B_{el} = B_{el}^{AB}(s) = 2b_A + 2b_B + 4s^e - 4.2$, with s given in units of GeV and B_{el} in GeV^{-2} . The constants $b_{A,B}$ are $b_P = 2.3$, $b_{p,r,w,j} = 1.4$, $b_{J/Y} = 0.23$. The increase of the slope parameter with c.m. energy is faster than the logarithmically one conventionally assumed; that way the ratio $\mathbf{s}_{el} / \mathbf{s}_{tot}$ remains well behaved at large energies. The diffractive cross sections are given by:

$$\frac{d\mathbf{s}_{sd(XB)}(s)}{dt dM^2} = \frac{g_{3P}}{16\mathbf{p}} \mathbf{b}_{AP} \mathbf{b}_{BP}^2 \frac{1}{M^2} \exp(B_{sd(XB)}t) F_{sd},$$

$$\frac{d\mathbf{s}_{sd(AX)}(s)}{dt dM^2} = \frac{g_{3P}}{16\mathbf{p}} \mathbf{b}_{AP}^2 \mathbf{b}_{BP} \frac{1}{M^2} \exp(B_{sd(AX)}t) F_{sd},$$

$$\frac{d\mathbf{s}_{dd}(s)}{dt dM_1^2 dM_2^2} = \frac{g_{3P}^2}{16\mathbf{p}} \mathbf{b}_{AP} \mathbf{b}_{BP} \frac{1}{M_1^2} \frac{1}{M_2^2} \exp(B_{dd}t) F_{dd}.$$

The couplings \mathbf{b}_{AP} are related to the Pomeron term $X^{AB}s^e$ of the total cross section parametrization. Picking a reference scale $\sqrt{s_{ref}} = 20 \text{ GeV}$, the couplings are given by $\mathbf{b}_{AP}\mathbf{b}_{BP} = X^{AB}s_{ref}^e$. The triple-Pomeron coupling is determined from single-diffractive data to be $g_{3P} \approx 0.318 \text{ mb}^{1/2}$; within the context of the formulæ in this section.

III-4) J/ψ production

Three main sources of J/ψ production may be distinguished:

- Decays of B mesons and baryons.
- Parton-shower evolution wherein a c and a \bar{c} quark produced in adjacent branchings turn out to have so small an invariant mass that the pair collapses to a single particle.
- Direct production, where a c quark loop gives a coupling between a set of gluons and a \bar{c} bound state. Higher-lying states, like the \mathbf{C}_c ones, may subsequently decay to J/ψ ; (this source, the 3rd one, is still a matter of controversy, with data overshooting the prediction from conventional sources).

In conclusion, The main conventional sources are:

- * $gg \rightarrow J/\psi g$ (i. e. process 86),
- * $gg \rightarrow \mathbf{C}_{0c} g$ (i. e. process 87),
- * $gg \rightarrow \mathbf{C}_{1c} g$ (i. e. process 88),
- * $gg \rightarrow \mathbf{C}_{2c} g$ (i. e. process 89).

The main processes for the 3rd source are processes 87, 88 and 89. In the program they are intended for applications at hadron colliders at non vanishing transverse momenta – in the limit of $P_T \rightarrow 0$ it is necessary to include $2 \rightarrow 1$ processes as well ($gg \rightarrow \mathbf{C}_{0c}$ i. e. process 104 and $gg \rightarrow \mathbf{C}_{2c}$ i. e. process 105) and to regularize divergences in the $2 \rightarrow 2$ graphs above.

IV) Practical informations

This chapter collects the informations I needed for building a sensible **pp physics** generator. In a second part, few examples are given on this subject (**pp physics**). Lastly Frequently Asked Questions (and answers) are listed and a **tutorial script interface to PYTHIA may be downloaded.**

IV-1) PYTHIA printed information and the event listing

To get the print of the full PYTHIA generated event (**the event-listing**) execute the FORTRAN code:

Call PYLIST(1)

(**LULIST(1) in PYTHIA 5.7.**) This gives an 80-column list of the partons/particles produced in the event: **the event-record**, i. e. some information stored in the common /PYJETS/ (**LUJETS/ in PYTHIA 5.7**):

```
REAL*8 P,V  
common/PYJETS/N,NPAD,K(4000,5),P(4000,5),V(4000,5)
```

(**the PYTHIA 5.7 event-record is:** common/LUJETS/N,K(4000,5),P(4000,5),V(4000,5) where P and V are in single precision; NPAD, a dummy integer in the PYTHIA 6.1 common block /PYJETS/ «align» the double-precision variables in the common).

In this common:

N number of partons/particles produced (number of the event-record lines).
In the following, the individual parton/particle number **,running from 1 to N is called i..**

NPAD is a dummy integer to «align» the real*8 variables P and V.

K(i,1) Status code **KS** of the parton/particle stored in line **i** (codes 1-10 correspond to currently existing partons/particles while larger codes contain partons/particles which no longer exist or other kinds of event-information:

- 0 Empty line.
- 1 Undecayed particle or unfragmented jet (single jet or the last of a jet system).
- 2 Unfragmented jet followed by more jets in the same colour-singlet jet system.
- 3 Unfragmented jet with special colour flow information stored in **K(i,4)** and **K(i,5)**, such that adjacent partons along the string need not follow each other in the event-record.
- 11 Decayed particle or fragmented jet (single jet or the last of a jet system, cf=1).
- 12 Fragmented jet followed by more jets in the same colour-singlet jet system, cf=2) Further , a B meson which decayed as a \bar{B} one, or vice-versa, because of $B-\bar{B}$ mixing, is marked with this code rather than 11.
- 13 Fragmented jet with special colour flow information.
- 14 Fragmented jet with special colour flow information.

21 Documentation line used to give a compressed story of the event at the beginning of the event-record.

- K(i,2)** Parton/particle **KF** code, as described in section **IV-2 b**.
K(i,3) Line number of the parent particle or jet where known, otherwise 0. Note that the assignment of a particle to a given jet in a jet system should not be taken too strictly, what is given there is only related to the way the event was generated.
K(i,4) Normally the line number of the first daughter; it is 0 for an undecayed particle or unfragmented jet. For **K(i,1)=3, 13 or 14**, instead it contains colour flow information for internal use only.
K(i,5) Normally the line number of the last daughter; it is 0 for an undecayed particle or unfragmented jet. For **K(i,1)=3, 13 or 14**, instead it contains colour flow information for internal use only.
P(i,1) P_x momentum (in GeV/c).
P(i,2) P_y momentum (in GeV/c).
P(i,3) P_z momentum (in GeV/c) (the Z axis is along the beam, see **IV-5 a** (FRAME) or the note hereafter).
P(i,4) E energy (in GeV).
P(i,5) m mass (in GeV/c^2), in parton showers, with space-like virtualities, i. e. where $Q^2 = -m^2 > 0$, one puts $P(i,5) = -Q$.
V(i,1) X position of the production vertex (in mm).
V(i,2) Y position of the production vertex (in mm).
V(i,3) Z position of the production vertex (in mm).
V(i,4) time of production (in $mm/c \approx 3.33 \times 10^{-12} s$).
V(i,5) Proper lifetime (in $mm/c \approx 3.33 \times 10^{-12} s$). If the particle is not expected to decay, **V(i,5)=0**. In the absence of electric or magnetic fields, or other disturbances, the decay vertex **VP** of an unstable particle may be calculated as: $VP(j) = V(i, j) + V(i, 5) * P(i, j) / P(i, 5)$ for $j=1-4$.

The event-listing might look as shown here (for a $pp \rightarrow h^0 \rightarrow gg$ (14 TeV) event) shortened in width (the last column «m» misses) to fill in a page:

Event listing (summary)								
I	particle/jet	KS	KF	orig	p_x	p_y	p_z	E
1	!p+	21	2212	0	.000	.000	7000.000	7000.000
2	!p+	21	2212	0	.000	.000	-7000.000	7000.000
3	!g!	21	21	1	-.190	-1.844	1260.221	1260.222
4	!g!	21	21	2	-.473	-.465	-129.761	129.763
5	!g!	21	21	3	.078	6.195	82.786	83.018
6	!g!	21	21	4	-6.414	.088	-29.812	30.495
7	!h0!	21	25	0	-6.335	6.282	52.974	113.512
8	!gamma!	21	22	7	-3.437	52.994	33.708	62.900
9	!gamma!	21	22	7	-2.899	-46.712	19.266	50.612
10	(h0)	11	25	7	-6.335	6.282	52.974	113.512
11	gamma	1	22	8	-3.437	52.994	33.708	62.900
12	gamma	1	22	9	-2.899	-46.712	19.266	50.612
13	(d)	A	12	1	1.019	-.837	-8.989	9.091
14	(g)	I	12	21	.445	-.460	-.527	.829
15	(g)	I	12	21	2.569	.830	-1.876	3.288

16	(g)	I	12	21	4	.939	1.054	-1.490	2.053
17	(g)	I	12	21	4	-.263	-.624	-.865	1.098
18	(g)	I	12	21	4	.014	-.249	.321	.407
19	(g)	I	12	21	4	3.858	1.719	.551	4.259
20	(g)	I	12	21	4	-.906	.351	-1.885	2.120
21	(g)	I	12	21	3	.194	-.379	5.213	5.230
22	(g)	I	12	21	3	-.846	-2.592	68.998	69.052
23	(g)	I	12	21	3	-.564	-.350	29.834	29.842
24	(g)	I	12	21	3	.180	.090	2.087	2.097
25	(g)	I	12	21	3	.561	.070	3.658	3.702
26	(uu_1)	V	11	2203	1	.380	.681	4229.159	4229.159
27	(dbar)	A	12	-1	4	.580	.829	-1.919	2.194
28	(g)	I	12	21	4	.605	.576	-.485	.966
29	(g)	I	12	21	4	-2.870	-3.369	-77.610	77.736
30	(d)	V	11	1	2	.047	.069	-239.515	239.516
...									
436	(K_S0)		11	310	330	.087	-.093	1.066	1.183
437	gamma		1	22	333	.152	.063	-.121	.205
438	gamma		1	22	333	.198	.160	-.416	.487
439	gamma		1	22	338	.178	.245	-1.498	1.528
440	gamma		1	22	338	-.023	.051	-.207	.215
441	gamma		1	22	342	.235	-.135	-16.694	16.696
442	gamma		1	22	342	-.003	.021	-1.238	1.238
443	gamma		1	22	345	-.083	-.041	-5.248	5.249
444	gamma		1	22	345	-.108	.102	-7.070	7.071
445	gamma		1	22	347	-.008	.002	-12.624	12.624
446	gamma		1	22	347	.093	-.091	-12.969	12.969
447	gamma		1	22	396	-.024	.107	-.160	.194
448	gamma		1	22	396	.047	-.014	-.144	.152
449	(pi0)		11	111	407	.222	1.128	25.159	25.185
450	(pi0)		11	111	407	.080	.422	5.684	5.702
451	pi-		1	-211	428	-.118	-.387	-.676	.801
...									
458	gamma		1	22	450	.008	.278	.2962	2.975
			sum:	2.00		.00	.00	.00	14000.01

Note: the 2 first lines are the 2 original protons and the lines 3 to 9 are a compressed summary of the event with the origin (column «orig») of the gammas at lines 8 and 9 being the h^0 (line 7). For the diquark (number 26) code, see the **note 1 IV-2-b**).

The «A» and «V» in the particle/jet column indicate the beginning and the end of a string (or cluster independent fragmentation) parton system. The «A», «I», and «V» give a «poor man» representation of an arrow. The columns E, m, p_x, p_y and p_z are self explanatory. All momenta, energy and masses are in GeV (since c=1). Note that **the Z axis plays the role of preferred direction along which the original protons are coming**. To lighten this summary 411 lines are skipped (from line 31 to line 435 and from line 452 to line 457). The final line is intended as a quick check that nothing weird happened (it contains the summed charge, summed momentum, summed energy and invariant mass) the values should agree with the input (the 2 first lines).. The double line after line 9 indicates the end of the summary (**the exclamation marks surrounding the particle names indicate they are part of the summary**). From line 10 downwards come the particle actually produced in the final states with a name **between parenthesis to indicate they have decayed**. Note that the origin of some particles could point to the same particle in the summary (gammas at lines 11 and 12 pointing to gammas at lines 8 and 9).

**Beware of this possible double counting when scanning the common /PYJETS/
(/LUJETS/ in PYTHIA 5.7)!**

IV-2) Particle codes list and particle decays table

IV-2 a) To get the **particle decays table**, execute these 11 lines of FORTRAN code:

	lines
program main	1
common/PYDAT1/mstu(200), paru(200), mstj(200), parj(200)	2
lout = 20 ! Listing file number.	3
C	4
C OPENING OF THE FILE 'decays.listing' on logical unit 'lout'..	5
C (that was why was necessary the presence of the common /PYDAT1/)	6
C	7
mstu(11) = lout ! Redirect PYTHIA output to the listing file.	8
Call PYLIST(12)	9
STOP	10
END	11

This will list on unit **lout** (file **decays.listing**) the full PYTHIA particle decays ↔ numbers relationship.

Note 1: «common /PYDAT1/» in PYTHIA 6.1 corresponds to «**common/LUDAT1/**» in **PYTHIA 5.7** Similarly, the subroutine «PYLIST» of PYTHIA 6.1 corresponds to the subroutine «**LULIST**» of **PYTHIA 5.7**

Note 2: One could have skipped the lines 2 to 8 (strictly speaking, lines 1, 9 and 11 are enough); when the program is run, with the simplest code (i. e. with only lines 1, 9 and 11), the output would have to be redirected the «UNIX way»: a.out>decays.listing. This piece of code as it is (lines 2 to 8) eases the further handling of PYTHIA output.

Note 3: The decay of a particle is switched «off» or «on» respectively by the FORTRAN statement:

Call PYGIVE('mdme(idc,1) = 0') ! Switch «off» the decay number idc.

Call PYGIVE('mdme(idc,1) = 1') ! Switch «on» the decay number idc.

mdme(idc,1) = -1 This a non-Standard Model decay mode, which by default is assumed to not exist. (see hereunder **note 5**).

mdme(idc,1) = 1 The decay channel **idc** is switched «on».

mdme(idc,1) = 0 The decay channel **idc** is switched «off».

mdme(idc,1) = 2 The decay channel **idc** is switched «on» for a particle but «off» for an antiparticle (it is also «on» for a particle its own antiparticle, i. e. it means the same than 1).

mdme(idc,1) = 3 The decay channel **idc** is switched «on» for an antiparticle but «off» for a particle (it is also «off» for a particle its own antiparticle).

Where **idc** is the decay number got with PYTLIST(12) (see the above-mentioned example-code line 9) and see the note of the section **III-1** related to the subroutine PYGIVE.

a) For a **quicker reference**, here are the numbers (**irel**) to add to the first decay number (210 in PYTHIA 6.1, 192 in PYTHIA 5.7) to get the decay number (**idc**) of the 14 h^0 standard decays:

irel	0	1	2	3	4	5	8	9	10	12	13	14	15	16
channel	$d\bar{d}$	$u\bar{u}$	$s\bar{s}$	$c\bar{c}$	$b\bar{b}$	$t\bar{t}$	e^+e^-	m^+m^-	t^+t^-	gg	gg	g^0	Z^0Z^0	W^+W^-

e. g. to switch **off** (see the above-mentioned **note 3**) the $h^0 \rightarrow gg$ decay channel (that is on by default in PYTHIA), you should add the FORTRAN statement:

```
call PYGIVE('mdme(223,1) = 0') ! 210+13=223
```

before PYTHIA initialization (see **IV-5 a**).

Note: In PYTHIA 5.7, «223» should be replaced by «205»

b) For a **quicker reference**, here are the numbers (**irel**) to add to the first decay number (174 in PYTHIA 6.1, 156 in PYTHIA 5.7) to get the decay number (**idc**) of the 12 Z^0 standard decays:

irel	0	1	2	3	4	5	8	9	10	11	12	13
channel	$d\bar{d}$	$u\bar{u}$	$s\bar{s}$	$c\bar{c}$	$b\bar{b}$	$t\bar{t}$	e^+e^-	$n_e\bar{n}_e$	m^+m^-	$n_m\bar{n}_m$	t^+t^-	$n_t\bar{n}_t$

e. g. to act (see the above-mentioned **note 3**) on the channel $Z^0 \rightarrow e^+e^-$, the **idc** number is 182 (174+ 8) in PYTHIA 6.1 (**164 in PYTHIA 5.7**).

Note 4: Hypothetical (i. e. non-Standard Model) h^0 decay channels (they are assumed by default not to exist) like channels **216, 217 and 221 in PYTHIA 6.1 or 198,199 and 203 in PYTHIA 5.7** ($b\bar{b}$, $t\bar{t}$ and t^+t^-) are set off with (see **note 5**) the FORTRAN statement:

```
Call PYGIVE('mdme(216,1) = -1') ! Hypoth. h0 decay channel switched «off».
```

This is also the case of the Hypothetical Z^0 decay channels **180, 181, 188 and 189 in PYTHIA 6.1 or 162, 163, 170 and 171 in PYTHIA 5.7** ($b\bar{b}$, $t\bar{t}$, t^+t^- and $n_t\bar{n}_t$).

Note 5: The two values **-1** and **0** may look similar, but in fact are quite different. In neither case the channel so set is generated, but in the latter case the channel still contributes to the total width of the resonance, and thus affects both simulated line shape and the generated cross section when PYTHIA is run. The value 0 is appropriate to a channel we assume exists, even if we are not currently simulating it, **while -1 should be used for channels we believe do not exist**. In particular, you are warned unwittingly to

set fourth generation channels 0 (**rather than -1**), since by now the support for a fourth generation is small.

IV-2 b) To get the briefer **particle codes list** (the particle code is the one referred in the PYTHIA event listing, see **IV-1**) execute the same FORTRAN code as in **IV-2 a** but with call **PYLIST(12)** (line 9) replaced by call **PYLIST(11)**.

For a **quicker reference**, the relationship particle names \leftrightarrow codes is given hereafter for the **quark and lepton codes** (an antiparticle, if existing, has the same negative code as its particle):

name	d	u	s	c	b	t	\tilde{b}	\tilde{t}	e^-	η_c	m^-	n_m	t^-	n_t	\tilde{t}	n_t
code	1	2	3	4	5	6	7	8	11	12	13	14	15	16	17	18

The **gauge boson and other fundamental boson codes** (an antiparticle has the same negative code as its particle):

name	g	\mathbf{g}	Z^0	W^+	h^0	Reggeon	Pomeron	Z^0
code	21	22	23	24	25	28	29	32

name	Z'^0	W'^+	H^0	A^0	H^+	\mathbf{h}_{techni}^0	L_Q	R^0
code	33	34	35	36	37	38	39	40

Note 1: A **meson** made up of a quark with code i and an antiquark with code $-j$, $j \neq i$, and with total spin s is given the code

$$KF = \{100\max(i, j) + 10\min(i, j) + 2s + 1\} \cdot \text{sign}(i - j) \cdot (-1)^{\max(i, j)}$$

A **diquark** made up of a quark with code i and another with code j where $i \geq j$, and with total spin s , is given a code $1000i + 100j + 2s + 1$

A **baryon** made up of quarks i , j and k , with $i \geq j \geq k$, and total spin s , is given a code $1000i + 100j + 10k + 2s + 1$

Note 2: When going from PYTHIA 5.7 to PYTHIA 6.1:

a) some particles have been renamed:

code **7:** from l to \tilde{b}

code **8:** from h to \tilde{t}

code **17:** from c to \tilde{t}

code **18:** from n_c to n_t

code **25:** from H^0 to h^0

code **35:** from H^0 to H^0

\tilde{t} , \tilde{b} , \tilde{t} and n_t denote a hypothetical fourth generation. Furthermore the antiparticle **that was noted in PYTHIA 5.7 with a tilde (~) is now** (to avoid confusion with supersymmetric particle names that begin with a tilde) **noted (in PYTHIA 6.1) with <bar> at the end of its name.**

b) 8 technicolor particles have been added to the one with code 38:

code 51: p_{techni}^0
 codes 52 and -52: p_{techni}^+ and p_{techni}^-
 code 53: $p_{techni}^{\prime 0}$
 code 54: r_{techni}^0
 codes 55 et -55: r_{techni}^+ et r_{techni}^-
 code 56: w_{techni}^0

c) some new particles have been added for doubly charged Higgs production in left-right-symmetric scenarios:

code 61 and -61: H_L^{++} and H_L^{--}
 code 62 and -62: H_R^{++} and H_R^{--}
 code 63 and -63: W_R^+ and W_R^-
 (the indices «L» and «R» indicate belonging to left or right SU(2) gauge group).

IV-3) Physics Processes numbering:

In the Table I of this section, PYTHIA processes are enumerated with the **ISUB** code that can be used to select desired processes via the FORTRAN statement

call PYGIVE('msub(isub) = 1')

(as shown in **III-1**). The processes are regrouped into **9** specific categories from **a**) to **i**). A wide selection of fundamental $2 \rightarrow 1$ and $2 \rightarrow 2$ tree processes of the Standard Model (electroweak and strong) has been included. In addition, a few «minimum bias» -type processes (like elastic scattering), loop graphs, box graphs, $2 \rightarrow 3$ tree graphs and some non-Standard Model processes are included. The classification is not always unique. A process that proceeds only via an s-channel state is classified as a $2 \rightarrow 1$ process (e. g. $q\bar{q} \rightarrow g^* / Z^0 \rightarrow e^+e^-$), but a $2 \rightarrow 2$ cross section may well have contributions from s-channel diagrams ($gg \rightarrow gg$ obtains contributions from $gg \rightarrow g^* \rightarrow gg$). Also, in the program, $2 \rightarrow 1$ and $2 \rightarrow 2$ graphs may sometimes be folded with two $1 \rightarrow 2$ splittings to form effective $2 \rightarrow 3$ or $2 \rightarrow 4$ processes ($W^+W^- \rightarrow h^0$ is folded with $q \rightarrow q'W^+$ and $\bar{q} \rightarrow \bar{q}''W^-$ to give $q\bar{q}' \rightarrow q''\bar{q} h^0$). In the following, f_i represents a fundamental fermion of flavour i , i. e. $d, u, s, c, b, t, b', t', e^-, n_e, \bar{m}, n_m, t^-, n_t, t'^-$ or n_t' . A corresponding antifermion is denoted \bar{f}_i . In several classes, some classes or fermions are explicitly excluded, since they do not couple to the g or the g^* (e. g. no $e^+e^- \rightarrow gg$). When processes have only been included for quarks, while leptons might also have been possible, the notation q_i is used. A lepton is denoted by l ; in a few cases neutrinos are also lumped under this heading. In processes where fermion masses are explicitly included in the matrix elements, an F is used to denote an arbitrary fermion and a Q a quark. Flavours appearing already in the initial state are denoted by indices i and j , whereas new flavours in the final state are denoted by k and l .

Charge-conjugate channels are always assumed included as well (where separate), and processes involving a W^+ also imply those involving a W^- . Wherever Z^0 is written, it is understood that g^* and g^* / Z^0 interference should be included as well (with

possibilities to switch off either, if so desired). In some cases this is not fully implemented, see further below. Correspondingly, Z^0 denotes the complete set $\mathbf{g}^* / Z^0 / Z^0$ (or some subset of it). Thus the notation \mathbf{g} is only used for a photon on the mass shell.

In the first column of Table I, a «+» denotes an implemented process while a blank is left for that is only foreseen. The second column gives the subprocess number **ISUB**, and third the description of the process. The final column gives references from which the cross sections have been obtained. Sometimes these references are to the original works on the subject, sometimes only to the place where the formulæ are given in the most convenient or accessible form. Apologies to all matrix-element calculators who are not mentioned. However, remember that this is not a review article on physics processes, but only a way for readers to know what is actually found in the program (these references are given after Table I).

Table I

In	No	Subprocess	reference
**	****	a) 2 → 1, tree	*****
+	1	$f_i \bar{f}_i \rightarrow \mathbf{g}^* / Z^0$	[Eic84]
+	2	$f_i \bar{f}_j \rightarrow W^\pm$	[Eic84]
+	3	$f_i \bar{f}_i \rightarrow h^0$	[Eic84]
	4	$\mathbf{g}W^\pm \rightarrow W^\pm$	
+	5	$Z^0 Z^0 \rightarrow h^0$	[Eic84, Cha85]
	6	$Z^0 W^\pm \rightarrow W^\pm$	
	7	$W^+ W^- \rightarrow Z^0$	
+	8	$W^+ W^- \rightarrow h^0$	[Eic84, Cha85]
**	****	b) 2 → 2, tree	*****
+	10	$f_i \bar{f}_j \rightarrow f_i \bar{f}_j$ (QFD)	[Ing87b]
+	11	$f_i \bar{f}_j \rightarrow f_i \bar{f}_j$ (QCD)	[Com77, Ben84, Eic84, Chi90]
+	12	$f_i \bar{f}_i \rightarrow f_k \bar{f}_k$	[Com77, Ben84, Eic84, Chi90]
+	13	$f_i \bar{f}_i \rightarrow gg$	[Com77, Ben84]
+	14	$f_i \bar{f}_i \rightarrow g\mathbf{g}$	[Hal78, Ben84]
+	15	$f_i \bar{f}_i \rightarrow gZ^0$	[Eic84]
+	16	$f_i \bar{f}_j \rightarrow gW^\pm$	[Eic84]
	17	$f_i \bar{f}_i \rightarrow gh^0$	
+	18	$f_i \bar{f}_i \rightarrow \mathbf{g}\mathbf{g}$	[Ber84]
+	19	$f_i \bar{f}_i \rightarrow \mathbf{g}Z^0$	[Eic84]
+	20	$f_i \bar{f}_j \rightarrow \mathbf{g}W^\pm$	[Eic84, Sam91]
	21	$f_i \bar{f}_i \rightarrow \mathbf{g}h^0$	
+	22	$f_i \bar{f}_i \rightarrow Z^0 Z^0$	[Eic84, Gun86]
+	23	$f_i \bar{f}_j \rightarrow Z^0 W^\pm$	[Eic84, Gun86]
+	24	$f_i \bar{f}_i \rightarrow Z^0 h^0$	[Ber85]

+	25	$f_i \bar{f}_i \rightarrow W^+ W^-$	[Bar94, Gun86]
+	26	$f_i \bar{f}_j \rightarrow W^\pm h^0$	[Eic84]
	27	$f_i \bar{f}_i \rightarrow h^0 h^0$	
+	28	$f_i g \rightarrow f_i g$	[Com77, Ben84]

Table I cont'd

In	No	Subprocess	reference
+	29	$f_i g \rightarrow f_i \mathbf{g}$	[Hal78,Ben84]
+	30	$f_i g \rightarrow f_i Z^0$	[Eic84]
+	31	$f_i g \rightarrow f_k W^\pm$	[Eic84]
	32	$f_i g \rightarrow f_i h^0$	
+	33	$f_i \mathbf{g} \rightarrow f_i g$	[Duk82]
+	34	$f_i \mathbf{g} \rightarrow f_i \mathbf{g}$	[Duk82]
+	35	$f_i \mathbf{g} \rightarrow f_i Z^0$	[Gab86]
+	36	$f_i \mathbf{g} \rightarrow f_k W^\pm$	[Gab86]
	37	$f_i \mathbf{g} \rightarrow f_i h^0$	
	38	$f_i Z^0 \rightarrow f_i g$	
	39	$f_i Z^0 \rightarrow f_i \mathbf{g}$	
	40	$f_i Z^0 \rightarrow f_i Z^0$	
	41	$f_i Z^0 \rightarrow f_k W^\pm$	
	42	$f_i Z^0 \rightarrow f_i h^0$	
	43	$f_i W^+ \rightarrow f_k g$	
	44	$f_i W^\pm \rightarrow f_k \mathbf{g}$	
	45	$f_i W^\pm \rightarrow f_k Z^0$	
	46	$f_i W^\pm \rightarrow f_k W^\pm$	
	47	$f_i W^\pm \rightarrow f_k h^0$	
	48	$f_i h^0 \rightarrow f_i g$	
	49	$f_i h^0 \rightarrow f_i \mathbf{g}$	
	50	$f_i h^0 \rightarrow f_i Z^0$	
	51	$f_i h^0 \rightarrow f_k W^\pm$	
	52	$f_i h^0 \rightarrow f_i h^0$	
+	53	$g g \rightarrow f_k \bar{f}_k$	[Com77,Ben84]
+	54	$g \mathbf{g} \rightarrow f_k \bar{f}_k$	[Duk82]
	55	$g Z^0 \rightarrow f_k \bar{f}_k$	
	56	$g W^\pm \rightarrow f_k \bar{f}_l$	
	57	$g h^0 \rightarrow f_k \bar{f}_l$	
+	58	$\mathbf{g} g \rightarrow f_k \bar{f}_k$	[Bar90]
	59	$\mathbf{g} Z^0 \rightarrow f_k \bar{f}_k$	
	60	$\mathbf{g} W^\pm \rightarrow f_k \bar{f}_l$	
	61	$\mathbf{g} h^0 \rightarrow f_k \bar{f}_k$	
	62	$Z^0 Z^0 \rightarrow f_k \bar{f}_k$	

Table I cont'd

In	No	Subprocess	reference
	63	$Z^0 W^\pm \rightarrow f_k \bar{f}_l$	
	64	$Z^0 h^0 \rightarrow f_k \bar{f}_k$	
	65	$W^+ W^- \rightarrow f_k \bar{f}_k$	
	66	$W^\pm h^0 \rightarrow f_k \bar{f}_l$	
	67	$h^0 h^0 \rightarrow f_k \bar{f}_k$	
+	68	$gg \rightarrow gg$	[Com77,Ben84]
+	69	$gg \rightarrow W^+ W^-$	[Kat83]
+	70	$gW^\pm \rightarrow Z^0 W^\pm$	[Kun87]
+	71	$Z^0 Z^0 \rightarrow Z^0 Z^0$ (longitudinal)	[Abb87]
+	72	$Z^0 Z^0 \rightarrow W^+ W^-$ (longitudinal)	[Abb87]
+	73	$Z^0 W^\pm \rightarrow Z^0 W^\pm$ (longitudinal)	[Dob91]
	74	$Z^0 h^0 \rightarrow Z^0 h^0$	
	75	$W^+ W^- \rightarrow gg$	
+	76	$W^+ W^- \rightarrow Z^0 Z^0$ (longitudinal)	[Ben87b]
+	77	$W^\pm W^\pm \rightarrow W^\pm W^\pm$ (longitudinal)	[Dun86,Bar90a]
	78	$W^\pm h^0 \rightarrow W^\pm h^0$	
	79	$h^0 h^0 \rightarrow h^0 h^0$	
+	80	$q_k g \rightarrow q_k p^\pm$	[Bag82]
**	****	c) $2 \rightarrow 2$, tree, massive final quarks	*****
+	81	$f_i \bar{f}_i \rightarrow Q_k \bar{Q}_k$	[Com79]
+	82	$gg \rightarrow Q_k \bar{Q}_k$	[Com79]
+	83	$q_i \bar{f}_j \rightarrow Q_k \bar{f}_l$	[Dic86]
+	84	$g\bar{g} \rightarrow Q_k \bar{Q}_k$	[Fon81]
+	85	$gg \rightarrow F_k \bar{F}_k$	[Bar90]
+	86	$gg \rightarrow J/\psi g$	[Bai83]
+	87	$gg \rightarrow c_{0c} g$	[Gas87]
+	88	$gg \rightarrow c_{1c} g$	[Gas87]
+	89	$gg \rightarrow c_{2c} g$	[Gas87]
+	106	$gg \rightarrow J/\psi g$	[Dre91]
+	107	$g\bar{g} \rightarrow J/\psi g$	[Ber81]
+	108	$gg \rightarrow J/\psi g$	[Jun97]
**	****	d) «minimum bias»	*****
+	91	elastic scattering	[Sch94]
+	92	single diffraction ($AB \rightarrow XB$)	[Sch94]
+	93	single diffraction ($AB \rightarrow AX$)	[Sch94]

Table I cont'd

In	No	Subprocess	Reference
+	94	double diffraction	[Sch94]
+	95	low- P_T production	[Sjö87]
**	****	e) $2 \rightarrow 1$, loop	*****
	101	$gg \rightarrow Z^0$	
+	102	$gg \rightarrow h^0$	[Eic84]
+	103	$gg \rightarrow h^0$	[Dre89]
**	****	f) $2 \rightarrow 2$, box	*****
+	110	$f_i \bar{f}_i \rightarrow gh^0$	[Ber85a]
+	111	$f_i \bar{f}_i \rightarrow gh^0$	[Ell88]
+	112	$f_i g \rightarrow f_i h^0$	[Ell88]
+	113	$gg \rightarrow gh^0$	[Ell88]
+	114	$gg \rightarrow gg$	[Con71, Ber84, Dic88]
+	115	$gg \rightarrow gg$	[Con71, Ber84, Dic88]
	116	$gg \rightarrow gZ^0$	
	117	$gg \rightarrow Z^0 Z^0$	
	118	$gg \rightarrow W^+ W^-$	
	119	$gg \rightarrow gg$	
**	****	g) $2 \rightarrow 3$, tree	*****
+	121	$gg \rightarrow Q_k \bar{Q}_k h^0$	[Kun84]
+	122	$q_i \bar{q}_i \rightarrow Q_k \bar{Q}_k h^0$	[Kun84]
+	123	$f_i \bar{f}_j \rightarrow f_i \bar{f}_j h^0$ (ZZ fusion)	[Cah84]
+	124	$f_i \bar{f}_j \rightarrow f_k \bar{f}_k h^0$ ($W^+ W^-$ fusion)	[Cah84]
+	131	$gg \rightarrow Z^0 Q_k \bar{Q}_k$	[Eij90]
**	****	h) non-Standard Model, $2 \rightarrow 1$	*****
+	141	$f_i \bar{f}_i \rightarrow g / Z^0 / Z^0$	[Alt89]
+	142	$f_i \bar{f}_j \rightarrow W^\pm$	[Alt89]
+	143	$f_i \bar{f}_j \rightarrow H^+$	[Gun87]
+	144	$f_i \bar{f}_j \rightarrow R$	[Ben85a]
+	145	$q_l \bar{l}_j \rightarrow L_Q$	[Wud86]
+	147	$dg \rightarrow d^*$	[Bau90]
+	148	$ug \rightarrow u^*$	[Bau90]
+	149	$gg \rightarrow h_{techni}^0$	[Eic84, App92]
+	151	$f_i \bar{f}_i \rightarrow H^0$	[Eic84]

Table I cont'd

In	No	Subprocess	Reference
+	152	$gg \rightarrow H^0$	[Eic84]
+	153	$\underline{gg} \rightarrow H^0$	[Dre89]
+	156	$f_i \bar{f}_i \rightarrow A^0$	[Eic84]
+	157	$gg \rightarrow A^0$	[Eic84]
+	158	$\underline{gg} \rightarrow A^0$	[Dre89]
**	****	i) non-Standard Model, $2 \rightarrow 2$ and $2 \rightarrow 3$	*****
+	161	$f_i g \rightarrow f_k H^+$	[Bar88]
+	162	$qg \rightarrow lL_Q$	[Hew88]
+	163	$gg \rightarrow L_Q \bar{L}_Q$	[Hew88,Eic84]
+	164	$q_i \bar{q}_i \rightarrow L_Q \bar{L}_Q$	[Hew88]
+	165	$f_i \bar{f}_i \rightarrow f_k \bar{f}_k$ (via \underline{g}^* / Z^0)	[Eic84,Lan91]
+	166	$f_i \bar{f}_j \rightarrow f_k \bar{f}_l$ (via W^\pm)	[Eic84,Lan91]
+	167	$qq \rightarrow q \bar{d}^*$	[Bau90]
+	168	$qq \rightarrow q \bar{u}^*$	[Bau90]
+	171	$f_i \bar{f}_i \rightarrow Z^0 H^0$	[Eic84]
+	172	$f_i \bar{f}_j \rightarrow W^\pm H^0$	[Eic84]
+	173	$f_i \bar{f}_j \rightarrow f_i \bar{f}_j H^0$ (ZZ fusion)	[Cah84]
+	174	$f_i \bar{f}_j \rightarrow f_k \bar{f}_l H^0$ ($W^+ W^-$ fusion)	[Cah84]
+	176	$f_i \bar{f}_i \rightarrow Z^0 A^0$	[Eic84]
+	177	$f_i \bar{f}_j \rightarrow W^\pm A^0$	[Eic84]
+	178	$f_i \bar{f}_j \rightarrow f_i \bar{f}_j A^0$ (ZZ fusion)	[Cah84]
+	179	$f_i \bar{f}_j \rightarrow f_k \bar{f}_l A^0$ ($W^+ W^-$ fusion)	[Cah84]
+	181	$gg \rightarrow Q_k \bar{Q}_k H^0$	[Kun84]
+	182	$q_i \bar{q}_i \rightarrow Q_k \bar{Q}_k H^0$	[Kun84]
+	186	$gg \rightarrow Q_k \bar{Q}_k A^0$	[Kun84]
+	187	$q_i \bar{q}_i \rightarrow Q_k \bar{Q}_k A^0$	[Kun84]

Note: New processes have been added (in PYTHIA 6.1, with respect to the list of processes of PYTHIA 5.7):

The processes 106, 107 and 108 (for J/ψ production) have been added, all of these are closely related to the existing process 86; only the colour- and charge-related prefactors differ in the matrix element expressions,

106: $gg \rightarrow J/\psi g$ **107:** $g\bar{g} \rightarrow J/\psi g$ **108:** $\underline{gg} \rightarrow J/\psi g$

- The list of (SUSY) processes and process numbers is according to tables 2 and 3 in the SPYTHIA manual (<http://www.thep.lu.se/~torbjorn/Pythia.html>)
- **Moreover 10 other new «photon» processes have been added** These 10 processes have been introduced for incoming virtual (spacelike) photons, as obtained e. g. in ep and

e^+e^- collisions. These are thus extensions of processes previously encoded for real photons:

- * $f_i \mathbf{g}_T^* \rightarrow f_i g$ (no 131, cf. process 33)
- * $f_i \mathbf{g}_L^* \rightarrow f_i g$ (no 132)
- * $f_i \mathbf{g}_T^* \rightarrow f_i \mathbf{g}$ (no 133, cf. process 34)
- * $f_i \mathbf{g}_L^* \rightarrow f_i \mathbf{g}$ (no 134)
- * $g \mathbf{g}_T^* \rightarrow f_i \bar{f}_i$ (no 135, cf. process 54)
- * $g \mathbf{g}_L^* \rightarrow f_i \bar{f}_i$ (no 136)
- * $\mathbf{g}_T^* \mathbf{g}_T^* \rightarrow f_i \bar{f}_i$ (no 137, cf. process 58)
- * $\mathbf{g}_T^* \mathbf{g}_L^* \rightarrow f_i \bar{f}_i$ (no 138)
- * $\mathbf{g}_L^* \mathbf{g}_T^* \rightarrow f_i \bar{f}_i$ (no 139)
- * $\mathbf{g}_L^* \mathbf{g}_L^* \rightarrow f_i \bar{f}_i$ (no 140)

Here the indices «T» and «L» represent transverse and longitudinal photons, respectively. In the limit of vanishing virtuality, the «T» photon cross section approaches that for a real photon, while the «L» one vanishes.

- **The process 99** $f_i \mathbf{g}^* \rightarrow f_i$ **has been added** for DIS scattering, by photon exchange only. Thus in this sense less powerful than process 10, but allows the use of the same photon flux machinery as for other $\mathbf{g}^* p$ and $\mathbf{g}^* \mathbf{g}^*$ processes,

- **Two new processes have been added for c_c production:**

$$gg \rightarrow c_{0c}$$
 (no 104)

$$gg \rightarrow c_{2c}$$
 (no 105).

(ref. [Bai83]). These are the lowest-order equivalents of processes 87 and 89. Note that $gg \rightarrow c_{1c}$ is forbidden, and so not included as a match to process 88.

- **Twenty-one technicolor processes have been added to the technicolor process 149:**

- $f_i \bar{f}_i \rightarrow \mathbf{r}_{techni}^0$ (no 191)
- $f_i \bar{f}_j \rightarrow \mathbf{r}_{techni}^+$ (no 192)
- $f_i \bar{f}_i \rightarrow \mathbf{w}_{techni}^0$ (no 193)
- $f_i \bar{f}_i \rightarrow f_k \bar{f}_k$ (no 194)
- $f_i \bar{f}_j \rightarrow f_k \bar{f}_i$ (no 195)
- $f_i \bar{f}_i \rightarrow W_L^+ W_L^-$ (no 361)
- $f_i \bar{f}_i \rightarrow W_L^\pm \mathbf{p}_{techni}^\mp$ (no 362)
- $f_i \bar{f}_i \rightarrow \mathbf{p}_{techni}^+ \mathbf{p}_{techni}^-$ (no 363)
- $f_i \bar{f}_i \rightarrow \mathbf{g}_{techni}^0$ (no 364)
- $f_i \bar{f}_i \rightarrow \mathbf{g}_{techni}'^0$ (no 365)
- $f_i \bar{f}_i \rightarrow Z^0 \mathbf{p}_{techni}^0$ (no 366)
- $f_i \bar{f}_i \rightarrow Z^0 \mathbf{p}_{techni}'^0$ (no 367)
- $f_i \bar{f}_i \rightarrow W^\pm \mathbf{p}_{techni}^\mp$ (no 368)
- $f_i \bar{f}_j \rightarrow W_L^\pm Z_L^0$ (no 370)
- $f_i \bar{f}_j \rightarrow W_L^\pm \mathbf{p}_{techni}^0$ (no 371)

- $f_i \bar{f}_j \rightarrow \mathbf{p}_{techni}^\pm Z_L^0$ (no 372)
- $f_i \bar{f}_j \rightarrow \mathbf{p}_{techni}^\pm \mathbf{p}_{techni}^0$ (no 373)
- $f_i \bar{f}_j \rightarrow \mathbf{g}_{techni}^\pm$ (no 374)
- $f_i \bar{f}_j \rightarrow Z^0 \mathbf{p}_{techni}^\pm$ (no 375)
- $f_i \bar{f}_j \rightarrow W^\pm \mathbf{p}_{techni}^0$ (no 376)
- $f_i \bar{f}_j \rightarrow W^\pm \mathbf{p}_{techni}^{\prime 0}$ (no 377)

The first 3 processes (**191**, **192** and **193**) are based on s-channel production of the respective resonance. All decay modes implemented can be simulated separately or in combination, in the standard fashion. These include pairs of fermions, or gauge bosons, or technipions, and of mixtures gauge bosons + technipions. Process **194** includes full interference between \mathbf{r}_{techni}^0 and \mathbf{w}_{techni}^0 . It can only be used for one final-state flavour at a time. This flavour is set in **KFPR(194,1)**. All of the processes from 361 to 377 can be accessed at once by setting **MSEL=50**.

-

- **Twelve new processes have been added to the production of doubly-charged Higgs** (in left-right-symmetric models, with an additional righthanded SU(2) gauge group):

- $l_i l_j \rightarrow H_L^{\pm\pm}$ (no 341)
- $l_i l_j \rightarrow H_R^{\pm\pm}$ (no 342)
- $l_i^\pm \mathbf{g} \rightarrow H_L^{\pm\pm} e^\mp$ (no 343)
- $l_i^\pm \mathbf{g} \rightarrow H_R^{\pm\pm} e^\mp$ (no 344)
- $l_i^\pm \mathbf{g} \rightarrow H_L^{\pm\pm} \mathbf{m}^\mp$ (no 345)
- $l_i^\pm \mathbf{g} \rightarrow H_R^{\pm\pm} \mathbf{m}^\mp$ (no 346)
- $l_i^\pm \mathbf{g} \rightarrow H_L^{\pm\pm} \mathbf{t}^\mp$ (no 347)
- $l_i^\pm \mathbf{g} \rightarrow H_R^{\pm\pm} \mathbf{t}^\mp$ (no 348)
- $f_i \bar{f}_i \rightarrow H_L^{++} H_L^{--}$ (no 349)
- $f_i \bar{f}_i \rightarrow H_R^{++} H_R^{--}$ (no 350)
- $f_i f_j \rightarrow f_k f_i H_L^{\pm\pm}$ (no 351)
- $f_i f_j \rightarrow f_k f_i H_R^{\pm\pm}$ (no 352)

-

new processes have been added for the production of Higgs pairs:

(these **Five** Higgs pair production have been added as explicit processes)

- $f_i \bar{f}_j \rightarrow H^\pm h^0$ (no 297)
- $f_i \bar{f}_j \rightarrow H^\pm H^0$ (no 298)
- $f_i \bar{f}_i \rightarrow A^0 h^0$ (no 299)

- $f_i \bar{f}_i \rightarrow A^0 H^0$ (no **300**)

- $f_i \bar{f}_i \rightarrow H^+ H^-$ (no **301**)

- Two new processes have been added to compositeness :

- $eg \rightarrow e^*$ (no **146**)

- $q_i \bar{q}_i \rightarrow e^\pm e^{*\mp}$ (no **169**)

(similar to existing processes **147**, **148** or **167**, **168** for q^*)

Warning: The process 21 should be noted 110 in Table I (erratum):

Table I References:

- [Abb87] A. Abbasabadi and W. Repko, Phys. Lett. **B199** (1987) 286; Phys. Rev. **D37** (1988) 2668;
W. Repko and G.L. Kane, private communication
- [Alt89] G. Altarelli, B. Mele and M. Ruiz-Altaba, Z. Phys. **C45** (1989) 109
- [App92] T. Appelquist and G. Triantaphyllou, Phys. Rev. Lett. **69** (1992) 2750
- [Bag82] J.A. Bagger and J.F. Gunion, Phys. Rev. **D25** (1982) 2287
- [Bai83] R. Baier and R. Rückl, Z. Phys. **C19** (1983) 251
- [Bar88] R.M. Barnett, H.E. Haber and D.E. Soper, Nucl. Phys. **B306** (1988) 697
- [Bar90] T.L. Barklow, SLAC-PUB-5364 (1990)
- [Bar90a] V. Barger, K. Cheung, T. Han and R.J.N. Phillips, Phys. Rev. **D42** (1990) 3052
- [Bar94] D. Bardin, M. Bilenky, D. Lehner, A. Olchevski and T. Riemann, Nucl. Phys. **B**, Proc. Suppl. **37B** (1994) 148;
D. Bardin, private communication
- [Bau90] U. Baur, M. Spira and P.M. Zerwas, Phys. Rev. **D42** (1990) 815
- [Ben84] H.-U. Bengtsson, Computer Physics Commun. **31** (1984) 323
- [Ben85a] H.-U. Bengtsson, W.-S. Hou, A. Soni and D.H. Stork, Phys. Rev. Lett. **55** (1985) 2762
- [Ben87b] M.C. Bento and C. H. Llewellyn Smith, Nucl. Phys. **B289** (1987) 36
- [Ber81] E.L. Berger and D. Jones, Phys. Rev. **D23** (1981) 1521
- [Ber84] E.L. Berger, E. Braaten and R.D. Field, Nucl. Phys. **B239** (1984) 52
- [Ber85] F.A. Berends and R. Kleiss, Nucl. Phys. **B260** (1985) 32
- [Ber85a] L. Bergström and G. Hulth, Nucl. Phys. **B259** (1985) 137
- [Cah84] R.N. Cahn and S. Dawson, Phys. Lett. **136B** (1984) 196;
R.N. Cahn, Nucl. Phys. **B255** (1985) 341;
G. Altarelli, B. Mele and F. Pitolli, Nucl. Phys. **B287** (1987) 205
- [Cha85] M. Chanowitz and M.K. Gaillard, Nucl. Phys. **B261** (1985) 379
- [Chi90] P. Chiappetta and M. Perrottet in «Large Hadron Collider Workshop», eds G. Jarlskog and D. Rein, CERN 90-10 (Geneva, 1990) Vol. II, p. 806
- [Com77] B.L. Combridge, J. Kripfganz and J. Ranft, Phys. Lett. **70B** (1977) 234;
R. Cutler and D. Sivers, Phys. Rev. **D17** (1978) 196
- [Com79] B.L. Combridge, Nucl. Phys. **B151** (1979) 429
- [Con71] V. Constantini, B. de Tollis and G. Pistoni, Nuovo Cim. **2A** (1971) 733
- [Dic86] D.A. Dicus and S.S.D. Willenbrock, Phys. Rev. **D34** (1986) 155
- [Dic88] D.A. Dicus and S.S.D. Willenbrock, Phys. Rev. **D37**(1988) 1801
- [Dob91] A. Dobado, M.J. Herrero and J.Terron, Z. Phys. **C50** (1991) 205, *ibid.* 465

- [Dre89] M. Drees, J. Ellis and D. Zeppenfeld, Phys. Lett. **B223** (1989) 454
- [Dre91] M. Drees and C.S. Kim, Z. Phys. **C53** (1991) 673
- [Duk82] D.W. Duke and J.F. Owens, Phys. Rev. **D26** (1982) 1600
- [Dun86] M.J. Duncan, G.L. Kane and W.W. Repko, Nucl. Phys. **B272** (1986) 517
- [Eic84] E. Eichten, I. Hinchliffe, K. Lane and C. Quigg, Rev. Mod. Phys. **56** (1984) 579; Rev. Mod. Phys. **58** (1985) 1065
- [Eij90] B. van Eijk and R. Kleiss, in «Large Hadron Collider Workshop», eds. G. Jarlskog and D. Rein, CERN 90-10 (Geneva, 1990), Vol. II, p. 183
- [Ell88] R.K. Ellis, I. Hinchliffe, M. Soldate and J.J. van der Bij, Nucl. Phys. **B297** (1988) 221
- [Fon81] M. Fontannaz, B. Pire and D. Schiff, Z. Phys. **C11** (1981) 211
- [Gab86] E. Gabrielli, Mod. Phys. Lett. **A1** (1986) 465
- [Gas87] R. Gastmans, W. Troost and T.T. Wu, Phys. Lett. **B184** (1987) 257
- [Gun86] J.F. Gunion and Z. Kunszt, Phys. Rev. **D33** (1986) 665;
errata as private communication from the authors
- [Gun87] J.F. Gunion, H.E. Haber, F.E. Paige, W.-K. Tung and S.S.D. Willenbrock, Nucl. Phys. **B294** (1987) 621
- [Hal78] F. Halzen and D.M. Scott, Phys. Rev. **D18** (1978) 3378
- [Hew88] J.L. Hewett and S. Pakvasa, Phys. Rev. **D37** (1988) 3165, and private communication from the authors
- [Ing87b] G. Ingelman et al., in «Proceedings of the HERA Workshop», ed. R.D. Peccei (DESY, Hamburg, 1988) Vol. 1, p. 3
- [Jun97] H. Jung, private communication;
H. Kharraziha, private communication
- [Kat83] M. Katuya, Phys. Lett. **124B** (1983) 421
- [Kun84] Z. Kunszt, Nucl. Phys. **B247** (1984) 339
- [Kun87] Z. Kunszt et al., in «Proceedings of the Workshop on Physics at Future Accelerators», ed. J.H. Mulvey, CERN 87-08 (1987), Vol. I, p. 123, and private communication
- [Lan91] K. Lane, private communication (1991)
- [Sam91] M. A. Samuel, G. Li, N. Sinha, R. Sinha and M.K. Sundaresan, Phys. Rev. Lett. **67** (1991) 9; ERRATUM *ibid.* 2920
- [Sch94] G.A. Schuler and T. Sjöstrand, Phys. Rev. **D49** (1994) 2257
- [Sjö87] T. Sjöstrand and M. Bengtsson, Computer Physics Commun. **43** (1987) 367
- [Wud86] J. Wudka, Phys. Lett. **167B** (1986) 337

IV-4) kinematics cuts (CKIN)

There are some kinematics cuts (**CKIN**) that can be set by you before the initialization (i. e. before the **PYINIT** call, see **IV-5 a**) and that affect the region of phase space within which events are generated. Some cuts are «hardwired» while most are «softwired». The hardwired ones are directly related to the kinematical variables used in the event selection procedure, and therefore have negligible effects on program efficiency. The most important of these are **CKIN(1) - CKIN(8)**, **CKIN(27) - CKIN(28)**, and **CKIN(31) - CKIN(32)**. The softwired ones are most of the remaining ones, that cannot be fully taken into account in the kinematical variable selection, so that generation in constrained regions of phase space may be slow. In extreme cases the phase space may be so small that the maximization procedure fails to find any allowed points at all (although some small region might still exist somewhere), and therefore switches off some subprocesses, or aborts altogether.

- CKIN(1), CKIN(2)** (default: 2. and -1. GeV) range of allowed $\hat{m} = \sqrt{\hat{s}}$ values. If **CKIN(2)**<0., the upper limit is inactive.
- CKIN(3), CKIN(4)** (default: 0. and -1. GeV) range of allowed \hat{P}_T values for hard $2 \rightarrow 2$ processes, with transverse momentum \hat{P}_T defined in the rest frame of the hard interaction. If **CKIN(4)**<0., the upper limit is inactive. For processes that are singular in the limit $\hat{P}_T \rightarrow 0$ (see **CKIN(6)**), **CKIN(5)** provides an additional constraint. The **CKIN(3)** and **CKIN(4)** limits can also be used in $2 \rightarrow 1 \rightarrow 2$ processes. Here, however, the product masses are not known and hence are assumed to be vanishing in the event selection. The actual P_T range for massive products is thus shifted downwards with respect to the nominal one.
- CKIN(5)** (default: 1. GeV) lower cut-off on \hat{P}_T values, in addition to the **CKIN(3)** cut above, for processes that are singular in the limit $\hat{P}_T \rightarrow 0$ (see **CKIN(6)**).
- CKIN(6)** (default: 1. GeV) hard $2 \rightarrow 2$ processes, which do not proceed only via an intermediate resonance (i. e. are $2 \rightarrow 1 \rightarrow 2$ processes), are classified as singular in the limit $\hat{P}_T \rightarrow 0$ if either or both of the two final-state products has a mass $m < \mathbf{CKIN(6)}$.
- CKIN(7), CKIN(8)** (default: -10. and 10.) range of allowed scattering subsystem rapidities $y = y^*$ in the c. m. frame of the event, where $y = (1/2)\ln(x_1/x_2)$.
- CKIN(9), CKIN(10)** (default -10. and 10.) range of allowed (true) rapidities for the product with largest rapidity in a $2 \rightarrow 2$ or a $2 \rightarrow 1 \rightarrow 2$ process, defined in the c.m. frame of the event, i.e. $\max(y_3^*, y_4^*)$. Note that rapidities are counted with sign, i. e. if $y_3^* = 1$ and $y_4^* = -2$ then $\max(y_3^*, y_4^*) = 1$.

- CKIN(11),CKIN(12)** (default -10. and 10.) range of allowed (true) rapidities for the product with smallest rapidity in a $2 \rightarrow 2$ or a $2 \rightarrow 1 \rightarrow 2$ process, defined in the c.m. frame of the event, i.e. $\min(y_3^*, y_4^*)$. Consistency thus requires **CKIN(11) ≤ CKIN(9)** and **CKIN(12) ≤ CKIN(10)**.
- CKIN(13),CKIN(14)** (default: -10. and 10.) range of allowed pseudorapidities for the product with largest pseudorapidity in a $2 \rightarrow 2$ or a $2 \rightarrow 1 \rightarrow 2$ process, defined in the c. m. frame of the event, i.e. $\max(\mathbf{h}_3^*, \mathbf{h}_4^*)$. Note that pseudorapidities are counted with sign, i. e. if $\mathbf{h}_3^* = 1$ and $\mathbf{h}_4^* = -2$ then $\max(\mathbf{h}_3^*, \mathbf{h}_4^*) = 1$.
- CKIN(15),CKIN(16)** (default: -10. and 10.) range of allowed pseudorapidities for the product with smallest pseudorapidity in a $2 \rightarrow 2$ or a $2 \rightarrow 1 \rightarrow 2$ process, defined in the c. m. frame of the event, i.e. $\min(\mathbf{h}_3^*, \mathbf{h}_4^*)$. Consistency thus requires **CKIN(15) ≤ CKIN(13)** and **CKIN(16) ≤ CKIN(14)**.
- CKIN(17),CKIN(18)** (default: -1. and 1.) range of allowed $\cos \mathbf{q}^*$ values for the product with largest $\cos \mathbf{q}^*$ value in a $2 \rightarrow 2$ or a $2 \rightarrow 1 \rightarrow 2$ process, defined in the c. m. frame of the event, i.e. $\max(\cos \mathbf{q}_3^*, \cos \mathbf{q}_4^*)$.
- CKIN(19),CKIN(20)** (default: -1. and 1.) range of allowed $\cos \mathbf{q}^*$ values for the product with smallest $\cos \mathbf{q}^*$ value in a $2 \rightarrow 2$ or a $2 \rightarrow 1 \rightarrow 2$ process, defined in the c.m. frame of the event, i. e. $\min(\cos \mathbf{q}_3^*, \cos \mathbf{q}_4^*)$. Consistency thus requires **CKIN(19) ≤ CKIN(17)** and **CKIN(20) ≤ CKIN(18)**.
- CKIN(21),CKIN(22)** (default: 0. and 1.) range of allowed x_1 values for the parton on side 1 that enters the hard interaction.
- CKIN(23),CKIN(24)** (default: 0. and 1.) range of allowed x_2 values for the parton on side 2 that enters the hard interaction.
- CKIN(25),CKIN(26)** (default: -1. and 1.) range of allowed Feynmann- x values, where $x_F = x_1 - x_2$.
- CKIN(27),CKIN(28)** (default: -1. and 1.) range of allowed $\cos \hat{\mathbf{q}}$ values in a hard $2 \rightarrow 2$ scattering, where $\hat{\mathbf{q}}$ is the scattering angle in the rest frame of the hard interaction.
- CKIN(31),CKIN(32)** (default: 2. and -1. GeV) range of allowed $\hat{m}' = \sqrt{\hat{s}'}$ values, where \hat{m}' is the mass of the complete three- or four-body final state in $2 \rightarrow 3$ or $2 \rightarrow 4$ processes (while \hat{m} , constrained in **CKIN(1)** and **CKIN(2)**, here corresponds to the one- or two-body central system). If **CKIN(32) < 0.**, the upper limit is inactive.
- CKIN(35),CKIN(36)** (default: 0. and -1. GeV^2) range of allowed $|\hat{t}| = -\hat{t}$ values in $2 \rightarrow 2$ processes. Note that for deep inelastic scattering this is

nothing but the Q^2 scale, in the limit that initial- and final-state radiation is neglected. If **CKIN(36)**<0., the upper limit is inactive.

CKIN(37),CKIN(38) (default: 0. and $-1. GeV^2$) range of allowed $|\hat{u}| = -\hat{u}$ values in $2 \rightarrow 2$ processes. If **CKIN(38)**<0., the upper limit is inactive.

CKIN(39),CKIN(40) (default: 4. and $-1. GeV^2$) the W^2 range allowed in DIS processes, i. e. subprocess number 10. If **CKIN(40)**<0., the upper limit is inactive. Here W^2 is defined in terms of $W^2 = Q^2(1-x)/x$. This formula is not quite correct, in that **i**) it neglects the target mass (for a proton), and **ii**) it neglects initial-state photon radiation off the incoming electron. It should be good enough for loose cuts however.

CKIN(41)-CKIN(44) (default: 12., -1., 12., -1. GeV) range of allowed mass values of the two (or one) resonances produced in a «true» $2 \rightarrow 2$ process, i. e. one not (only) proceeding through a single s-channel resonance ($2 \rightarrow 1 \rightarrow 2$). (These are the ones listed as $2 \rightarrow 2$ in Table I of IV-3) Only particles with a width above **PARP(41)** are considered as bona fide resonances and tested against the **CKIN** limits; particles with a smaller width are put on the mass shell without applying any cuts. The exact interpretation of the **CKIN** variables depends on the flavours of the two produced resonances.

For two resonances like $Z^0 W^+$ (produced from $ff' \rightarrow Z^0 W^+$), which are not identical and which are not each other's antiparticles, one has

CKIN(41)< m_1 <**CKIN(42)**, and

CKIN(43)< m_2 <**CKIN(44)**,

where m_1 and m_2 are the actually generated masses of the two resonances, and 1 and 2 are defined by the order in which they are given in the production process specification.

For two resonances like $Z^0 Z^0$, which are identical, or $W^+ W^-$, which are each other's antiparticles, one instead has

CKIN(41)< $\min(m_1, m_2)$ <**CKIN(42)**, and

CKIN(43)< $\max(m_1, m_2)$ <**CKIN(44)**.

In addition, whatever limits are set on **CKIN(1)** and, in particular, on **CKIN(2)** obviously affect the masses actually selected.

Note 1: If **MSTP(42)**=0, so that no mass smearing is allowed, the **CKIN** values have no effect (the same as for particles with too narrow a width).

Note 2: If **CKIN(42)**<**CKIN(41)** it means that the **CKIN(42)** limit is inactive, correspondingly, if **CKIN(44)**<**CKIN(43)** then **CKIN(44)** is inactive.

Note 3: If limits are active and the resonances are identical, it is up to you to ensure that **CKIN(41)**≤**CKIN(43)** and **CKIN(42)**≤**CKIN(44)**.

Note 4: For identical resonances, it is not possible to preselect which of the resonances is the lighter one; if, for instance, one Z^0 is to decay to leptons and the other to quarks, there is no mechanism to guarantee that the lepton pair has a mass smaller than the quark one.

Note 5: The **CKIN** values are applied to all relevant $2 \rightarrow 2$ processes equally, which may not be what one desires if several processes are generated simultaneously. Some caution is therefore urged in the use of the **CKIN(41)-CKIN(44)** values. Also in other respects, users are recommended to take proper care: if a Z^0 is only allowed to decay into $b\bar{b}$, for example, setting its mass range to be 2-8 *GeV* is obviously not a good idea.

Note 6: In principle, the machinery should work for any $2 \rightarrow 2$ process with resonances in the final state, but so far it has only been checked for processes 22-26, so also from this point some caution is urged.

CKIN(45)-CKIN(48) (default: 12., -1., 12., -1. *GeV*) range of allowed mass values of the two (or one) secondary resonances produced in a $2 \rightarrow 1 \rightarrow 2$ process (like $gg \rightarrow h^0 \rightarrow Z^0 Z^0$) or even a $2 \rightarrow 2 \rightarrow 4$ (or 3) process (like $q\bar{q} \rightarrow Z^0 h^0 \rightarrow Z^0 W^+ W^-$). Note that these **CKIN** values only affect the secondary resonances; the primary ones are constrained by **CKIN(1)**, **CKIN(2)** and **CKIN(41)-CKIN(44)** (indirectly, of course, the choice of primary resonance masses affects the allowed mass range for the secondary ones). What is considered to be a resonance is defined by **PARP(41)**; particles with a width smaller than this are automatically put on the mass shell. The description closely parallels the one given for **CKIN(41) - CKIN(44)**. Thus, for two resonances that are not identical or each other's antiparticles, one has **CKIN(45) < m_1 < CKIN(46)**, and **CKIN(47) < m_2 < CKIN(48)**,

where m_1 and m_2 are the actually generated masses of the two resonances, and 1 and 2 are defined by the order in which they are given in the decay channel specification in the program (see e. g. output from **PYSTAT(2)** or **PYLIST(12)**). For two resonances that are identical or each other's antiparticles, one instead has **CKIN(45) < $\min(m_1, m_2)$ < CKIN(46)**, and **CKIN(47) < $\max(m_1, m_2)$ < CKIN(48)**.

Note 1-5: as for **CKIN(41) - CKIN(44)**, with trivial modifications.

Note 6: Setting limits on secondary resonance masses is possible in any of the channels of the allowed types (see above). However, so far only $h^0 \rightarrow Z^0 Z^0$ and $h^0 \rightarrow W^+ W^-$ have been fully implemented, such that an arbitrary mass range below the naïve mass threshold may be picked. For other possible resonances, any restrictions made on the allowed mass range are not reflected in

the cross section; and further it is not recommendable to pick mass windows that make a decay on the mass shell impossible. These limitations will be relaxed in future versions.

CKIN(51)-CKIN(56) (default: 0., -1. 0., -1., 0., -1. *GeV*) range of allowed transverse momenta in a true $2 \rightarrow 3$ process. Currently two different alternatives are around. For subprocess 131, the P_T of the first product (the Z^0) is set by **CKIN(3)** and **CKIN(4)**, while for the quark and antiquark P_T 's one has

CKIN(51) < $\min(P_{Tq}, P_{T\bar{q}})$ < **CKIN(52)**, and

CKIN(53) < $\max(P_{Tq}, P_{T\bar{q}})$ < **CKIN(54)**.

Negative **CKIN(52)** and **CKIN(54)** values means that the corresponding limits are inactive. For subprocesses 121-124, and their H^0 and A^0 equivalents (173, 174, 178, 179, 181, 182, 186, 187), **CKIN(51) – CKIN(54)** again corresponds to P_T ranges for scattered partons, but in order of appearance, i. e. **CKIN(51)-CKIN(52)** for the parton scattered off the beam and **CKIN(53) – CKIN(54)** for the one scattered off the target. **CKIN(55)** and **CKIN(56)** here sets P_T limits for the third product, the h^0 , i. e. the **CKIN(3)** and **CKIN(4)** values have no effect for this process. Since the P_T of the Higgs is not one of the primary variables selected, any constraints here may mean reduced Monte Carlo efficiency, while for these processes **CKIN(51) – CKIN(54)** are «hardwired» and therefore do not cost anything.

Note: New CKIN cuts (CKIN(61) to CKIN(78)) have been introduced in PYTHIA 6.1 to restrict the range of kinematics for the photons generated off the lepton beams. In each quartet of numbers, the first two corresponds to the range allowed on incoming side 1 (beam) and the last two to side 2 (target). The cuts are only applicable for a lepton beam. Note that the x and Q^2 (P_2) variables are the basis for the generation, and so can be restricted with no loss of efficiency. For leptonproduction the W is uniquely given by the one x value of the problem, so here also W cuts are fully efficient. The other cuts may imply a slowdown of the program, but not as much as if equivalent cuts only are introduced after events are fully generated.

This machinery cannot be combined with the variable-energy option obtainable for **MSTP(171)=1**. The reason is that a variable-energy machinery is now used internally for the gamma-hadron or gamma-gamma subsystem, with some information saved at initialization for the full energy.

CKIN(61)-CKIN(64) (default: 0.0001, 0.99, 0.0001, 0.99) allowed range for the energy fractions x that the photon take of the respective incoming lepton energy. These fractions are defined in the cm frame of the collision, and differ from energy fractions as defined in another frame. (Watch out at HERA!). In order to avoid some technical problems, absolute lower and upper limits are set internally at 0.0001 and 0.9999.

- CKIN(65)-CKIN(68)** (default: 0., -1., 0., -1. GeV^2) allowed range for the spacelike virtuality of the photon, conventionally called either Q2 or P2, depending on process. A negative number means that the upper limit is inactive, i. e. purely given by kinematics. A nonzero lower limit is implicitly given by kinematics constraints.
- CKIN(69)-CKIN(72)** (default: 0., -1., 0., -1.) allowed range of the scattering angle theta of the lepton, defined in the cm frame of the event. (**Watch out at HERA!**). A negative number means that the upper limit is inactive, i. e. equal to pi.
- CKIN(73)-CKIN(76)** (default: 0.0001, 0.99, 0.0001, 0.99) allowed range for the lightcone fraction y that the photon take of the respective incoming lepton energy. The lightcone is defined by the four-momentum of the lepton or hadron on the other side of the event (and thus deviates from true lightcone fraction by mass effects that normally are negligible). The y value is related to the x and Q2 (P2) values by $y = x + Q2/s$ if mass terms are neglected.
- CKIN(77), CKIN(78)** (default: 2., -1. GeV) allowed range for W, i. e. either the photon-hadron or photon-photon invariant mass. A negative number means that the upper limit is inactive.

IV-5) Initializing PYTHIA and random sequences seed

a) Initializing PYTHIA:

To initialize PYTHIA, execute (after having set the generation condition with the PYTHIA variables definition as shown in the examples) the FORTRAN statement:

Call PYINIT(FRAME,BEAM,TARGET,WIN)

With:

- FRAME** Character variable used to specify the frame of the experiment:
 «CMS» indicates a collider beam experiment with beam pointing in + Z direction and target in - Z direction),
 «FIXT» indicates a fixed-target experiment with beam pointing in + Z direction),
 «USER» specifies user's frame by giving beam momentum in **P(1,1)**, **P(1,2)** and **P(1,3)** and target momentum in **P(2,1)**, **P(2,2)** and **P(2,3)** in common block /PYJETS/ (**LUJETS/ for PYTHIA 5.7** see **IV-1**)
 Particles are assumed on the mass shell and energies are calculated accordingly.
- BEAM** Character variable to specify the beam particle (see hereunder).
- TARGET** Character variable to specify the target particle:
 «e-», «e+», «nu_e», «nu_ebar», «mu-», «mu+», «nu_mu», «nu_mubar»,
 «tau-», «tau+», «nu_tau», «nu_taubar», «gamma», «pi0», «pi+», «pi-»,
 «n0», «nbar0», «p+», «pbar-», «Lambda0», «Sigma-», «Sigma0»,
 «Sigma+», «Xi-», «Xi0», «Omega-».
- WIN** Energy of system that depends from the **frame** chosen by user:
 «CMS»: Total energy of system (in GeV),

«FIXT»: Momentum of beam particle (in GeV/c),
 «USER»: dummy (as information is taken from the above-mentioned **P** vectors).

e.g. to initialize a **pp collision at 14 TeV (in a collider)**:

```
call PYINIT('CMS', 'p', 'p', 1.4d+4)
```

b) Random sequences seed:

The generator has a period of over 10^{43} , and the possibility to obtain almost 10^9 different and disjoint subsequences, selected by giving an initial integer number.

The price to pay for such a long period is that the state of the generator (the state of the random sequences) at a given moment cannot be described by a single integer but requires about 100 words (most of these are real numbers).

As PYTHIA is a heavy user of random numbers, the normal procedure, i. e. restart the generation with a seed value written to the run output (at the run end), would have been time consuming (on random number administration)..

The PYTHIA solution is to choose which of the possible subsequences will be initialized in the next random generator call with a given value. This is done by setting **MRPY(1)** (**MRLU(1) in PYTHIA 5.7**) to the chosen subsequence number i. e. a number between 1 and 899,999,999 (i. e. that warrants 900,000,000 different generations). Thus, if you want to have always the same generation, you should set (before the call to PYINIT) MRPY(1) to always the same value as:

```
Call PYGIVE('mrpy(1) = 1019290') ! Random sequences seed.
```

1019290 is an example (as the default is to obtain the same sequence in each run, doing nothing is enough). This number could just as good as **123456789** or any number between 1 and 899,999,999.

To be sure that every generation will be different, an elegant solution (unfortunately machine dependent) is to link the seed to the time-stamp (with sec.) of the machine on which PYTHIA is running e.g.:

```
character*8 ctime,ctemp,ctempf
character arg*18
integer i,j
```

```
C
C GET THE TIME STAMP 'ctime' OF THE MACHINE (8 CHAR WITH DOTS)
C THE FOLLOWING CODE WILL CONVERT THE TIME-STAMP TO A..
C ..8 DIGITS NUMBER.
C
```

```
  j = 0
  ctemp = '      '
  do i = 1,8 ! Remove dots from 'ctime'.
  if(ctime(i:i).ne.'.')then
    j = j + 1
    ctemp(j:j) = ctime(i:i)
  endif ! ctime(i:i)<>'.'
  enddo ! i-loop.
```



```

ctemp(1:8) = ctemp(1:6)//'00' ! Fillup to 8 char.
arg(1:18) = 'mrpy(1) '//ctemp(1:8)
call PYGIVE(arg(1:18)) ! Random sequences seed.

```

Unfortunately, this link of the random seed to the time-stamp of the machine (ctime) is machine-dependent. (this code is given as an example).

IV-6) Examples of pp physics generation

The collision (in some semblance of a time order) is described in the beginning of the section **II**. The proton beam PYTHIA handling is explained in the first paragraph of the section **II-5**.

IV-6-1) $pp \rightarrow h^0 \rightarrow gg$ generator:

The following 69 lines of FORTRAN code would generate 10,000 $h^0 \rightarrow gg$ events in a **pp collider**:

	lines
program main	0
external pydata	1
common/PYDAT1/mstu(200),paru(200),mstj(200),parj(200)	2
integer i,j,lout,nevents,iev	3
character*8 cdat,ctime,ctemp,ctempf	4
character arg*18	5
lout = 20 ! Listing file log. unit number.	6
C	7
C OPENING OF THE FILE 'pythia.listing' ON LOGICAL UNIT 'lout'	8
C (that was why was necessary the presence of the common /PYDAT1/)	9
C	10
mstu(11) = lout ! Redirect PYTHIA output to the 'pythia.listing' file.	11
C*****h0 PRODUCTION (4 next lines)*****	12
call PYGIVE('msel = 0') ! Full user control.	13
call PYGIVE('msub(102) = 1') ! gluon fusion (gg->h0).	14
call PYGIVE('msub(123) = 1') ! Z0Z0 fusion (fi_fj->fi_fj_h0).	15
call PYGIVE('msub(124) = 1') ! W+W- fusion (fi_fj->fk_fl_h0).	16
C*****h0 STANDARD DECAYS (14 next lines)*****	17
call PYGIVE('mdme(210,1) = 0') ! h0->d_dbar off.	18
call PYGIVE('mdme(211,1) = 0') ! h0->u_ubar off.	19
call PYGIVE('mdme(212,1) = 0') ! h0->s_sbar off.	20
call PYGIVE('mdme(213,1) = 0') ! h0->c_cbar off.	21
call PYGIVE('mdme(214,1) = 0') ! h0->b_bbar off.	22
call PYGIVE('mdme(215,1) = 0') ! h0->t_tbar off.	23
call PYGIVE('mdme(218,1) = 0') ! h0->e+e- off.	24
call PYGIVE('mdme(219,1) = 0') ! h0->mu+mu- off.	25
call PYGIVE('mdme(220,1) = 0') ! h0->tau+tau- off.	26
call PYGIVE('mdme(222,1) = 0') ! h0->gg off.	27
call PYGIVE('mdme(224,1) = 0') ! h0->gamma_Z0 off.	28
call PYGIVE('mdme(225,1) = 0') ! h0->Z0_Z0 off.	29
call PYGIVE('mdme(226,1) = 0') ! h0->W+W- off.	30

```

        call PYGIVE('mdme(223,1) = 1') ! h0->gamma_gamma on.           31
C*****h0 NON-STANDARD DECAYS (involving a fourth generation)      32
C*****(-1 switch them off , 3 next lines)*****                  33
        call PYGIVE('mdme(216,1) = -1') ! h0->b'_b'bar off.         34
        call PYGIVE('mdme(217,1) = -1') ! h0->t'_t'bar off.         35
        call PYGIVE('mdme(221,1) = -1') ! h0->tau'+_tau'- off.      36
C*****MASSES (given as examples, 3 next lines)*****              37
        call PYGIVE('pmas(6,1) = 175.') ! t.                          38
        call PYGIVE('pmas(23,1) = 91.187.') ! Z0.                     39
        call PYGIVE('pmas(25,1) = 100.') ! h0.                         40
C                                                                      41
C LINK THE RANDOM SEED TO THE MACHINE TIME STAMP                     42
C                                                                      43
        ctemp = '          '                                          44
        call DATIMH(cdat,ctime) ! Get the machine time-stamp.         45
        j=0                                                            46
        do i = 1,8 ! Remove the dots from 'ctime'.                    47
        if(ctime(i:i).ne.'.')then                                       48
            j=j + 1                                                    49
            ctemp(j:j) = ctime(i:i)                                     50
        endif ! ctime(i:i)<>'.'                                         51
        enddo ! i-loop.                                                52
        ctempf(1:8) = ctemp(1:6)//'00' 1 Fillup to 8 char.            53
        arg(1:18) = 'mrpy(1) = '//ctempf(1:8)                          54
C*****INITIALIZATION (2 next lines)*****                          55
        call PYGIVE(arg(1:18)) ! Random sequences seed.                56
        call PYINIT('CMS', 'p', 'p', 1.4d+4) ! LHC collisions.        57
C LOOP ON 'nevents' (10,000 for the example) GENERATED EVENTS      58
        nevents = 10000                                               59
        do iev = 1,nevents                                             60
            call PYEVNT ! Generate one PYTHIA event.                   61
            if(iev.eq.1)call PYLIST(1) ! PYTHIA detail of the 1st event. 62
C THE FOLLOWING ROUTINE (at line 65) SHOULD BE PROVIDED..           63
C ..BY USER TO PICKUP THE PYTHIA INFORMATION HE NEEDS.             64
        call PREVNT ! Processes the event following user's wish.      65
        enddo ! iev-loop.                                             66
        CLOSE(lout)                                                    67
        STOP                                                            68
        END                                                            69

```

Note: the “external pydata” in the line number 1 **is mandatory with PYTHIA 6.1 With PYTHIA 5.7 it should be external ludata.** (as pydata and ludata are BLOCK DATA subprograms which are not automatically included by many Unix linkers, including alpha, Linux and SGI. There are several techniques for getting them linked, but using the external statement is one of the simplest which works on most).

See **IV-2** and notes **4** and **5** for switching «off» the h^0 hypothetical decays (lines 34 to 36).

IV-6-2) $pp \rightarrow$ PYTHIA default min. bias and CMS recommended min. bias:

a) The PYTHIA default minimum bias includes the elastic and diffractive topologies (low P_T events) unless user increases **CKIN(3)** (see **IV-4**) so as to obtain high- P_T jets (see the note).

The FORTRAN code for generating the PYTHIA default min. bias is very simple: Replace the lines 12 to 40 of the above-mentioned example (**IV-6-1**) by nothing (i. e. let the default **MSEL=1**).

b) The recommended CMS minimum bias is set by replacing the lines 12 to 40 of the above-mentioned example (**IV-6-1**) by the 6 following FORTRAN statements:

	lines
call PYGIVE('msel = 1') ! pp min. bias.	1
call PYGIVE('mstp(81) = 1') ! multiple interaction on.	2
call PYGIVE('mstp(82) = 4') ! impact parameter choice.	3
call PYGIVE('mstp(2) = 2') ! 2 nd order running of alpha_s.	4
call PYGIVE('mstp(33) = 3') ! K-factors.	5
call PYGIVE('parp(82) = 3.2') ! PT0 multiple distribution tail.	6

The line number 3 requires multiple interactions assuming a varying impact parameter and a hadronic matter overlap consistent with a double Gaussian matter distribution given by **PARP(83)** and **PARP(84)**, with a continuous turn-off of the cross section at $P_{T0} = \mathbf{PARP(82)}$, the line number 6 defines the regularization scale P_{T0} of the transverse momentum spectrum for multiple interactions (with **MSTP(82)** ≥ 2).

Important note: These 2 minimum bias settings do not take into account the multiple interactions (pileup) always present in bunch crossing with intense beams. This pileup has thus to be done on user's responsibility (all the more as the pileup facility that comes with PYTHIA is of little use since the event record common block is too small (4000) to contain tens of full-fledged LHC events).

IV-6-3) $pp \rightarrow q\bar{q}$:

To generate quark-antiquark production in a **pp** collision you should replace the lines 12 to 40 of the above-mentioned example (**IV-6-1**) either by:

```
call PYGIVE('msel = 4') (for  $pp \rightarrow c\bar{c}$ ) or
call PYGIVE('msel = 5') (for  $pp \rightarrow b\bar{b}$ ) or
call PYGIVE('msel = 6') (for  $pp \rightarrow t\bar{t}$ ) or
call PYGIVE('msel = 7') (for  $pp \rightarrow b\bar{b}$ ) or
call PYGIVE('msel = 8') (for  $pp \rightarrow t\bar{t}$ ).
```

Note: For charm and bottom additional sources of production coming from flavour excitation and parton shower branchings, make the picture more complicated.

IV-6-4) $pp \rightarrow$ prompt photons:

To generate prompt photons, you should replace the lines 12 to 40 of the above-mentioned example (**IV-6-1**) by:

```
call PYGIVE('msel=10')
```

You should also set $P_{T\min}$ in **CKIN(3)** (see **IV-4**).

IV-6-5) $pp \rightarrow Z^0$ or $pp \rightarrow Z^0 + jet$:

The Z^0 and $Z^0 + jet$ production in a **pp** collision may be generated by replacing the lines 12 to 40 of the above-mentioned example (**IV-6-1**) either by:

call PYGIVE('msel=11') (for a Z^0 production) or
call PYGIVE('msel=13') (for a $Z^0 + jet$ production)

In these two generations you may choose the Z^0 decay channel by mimicking the above-mentioned **IV-6-1** example (lines 17 to 36) with the value of the first argument **idc** of the array **mdme(idc,1)** taken in **IV-2-a b**).

IV-6-6) $pp \rightarrow W^\pm$ or $pp \rightarrow W^\pm + jet$:

The W^\pm and $W^\pm + jet$ production in a **pp** collision may be generated by replacing the lines 12 to 40 of the above-mentioned example (**IV-6-1**) either by:

call PYGIVE('msel = 12') (for a W^\pm production) or
call PYGIVE('msel=14') (for a $W^\pm + jet$ production).

IV-6-7) $pp \rightarrow h^0 Z^0$ or $pp \rightarrow h^0 W^\pm$:

The $h^0 Z^0$ or $h^0 W^\pm$ production in a **pp** collision may be generated by replacing the lines 12 to 40 of the above-mentioned example (**IV-6-1**) by:

call PYGIVE('msel=17')

Note 1: The h^0 decay channel may be chosen as shown in the **IV-6-1** above-mentioned example (lines 17 to 36).

Note 2: As either $h^0 Z^0$ or $h^0 W^\pm$ will be produced (according to the cross section) it is left to user's responsibility to select the chosen process with the proper «**isub**» code in **MSTI(1)** (according to **Table I** of **IV-3**, **isub=24** for $h^0 Z^0$ and **26** for $h^0 W^\pm$) knowing that «**MSTI**» is in the common:

common/PYPARS/mstp(200),parp(200),msti(200),pari(200)

e. g. the test should be done on **MSTI(1) = 24** for a $h^0 Z^0$ event per event selection.

Note: with **PYTHIA 6.1** the common/PYPARS/ should be preceded by the declaration:

real*8 parp,pari

IV-7) Frequently asked questions:

1

-Is the event-listing representing the event as a «perfect» detector would see it?

Answer: yes, the event-listing should be representative of the way the event should look if no detector were there to mess it up.

2

-What is the unit in the cross section table once PYTHIA is initialized?

Answer: Cross sections units are always mb. More interesting are the final cross sections; the maximums are mainly for internal Monte Carlo use.

3

-Why a run with MSEL=1 and another with MSEL=0 plus the processes 11, 12, 13, 28, 53 and 68 switched 'on' give different results?

Answer: MSEL=1 by default includes low-P_T events, unless you increased CKIN(3) so as to only obtain high-P_T jets. I refer to the manual, where it says that for MSEL=1 also MSUB(95) is 'on' for CKIN(3)<PARP(81), which is the case if you did not do anything. If you set a CKIN(3)>PARP(81), the two runs should agree. Such a minimum P_T cut you are anyway forced to provide when studying jet production. (Even the default P_Tmin=CKIN(3)=0 is saved by the further cut in CKIN(5)=1 GeV, which is there as an ultimate check against singular behaviour).

4

-I want to select isolated gammas from all the stuff that can spoil physics on this subject (quark brems etc.). Am I right if I find an experimental way to reject i) the decays of pi^0, eta^0, eta' & omega and ii) the processes 14, 18, 29 and 96?

Answer: Among the 'classic' backgrounds also figures 114 and 115. With process 96 I presume you mean the bremsstrahlung photons in 'minimum bias events', e. g. obtained with MSEL=1 and no special cuts (or some P_Tmin cut in CKIN(3) for high-P_T physics). Then it should be OK. Of course if there are also other requirements, like leptons, then other processes cannot be neglected (19 → gamma+lepton pair, e. g.).

5

-I want to generate pp->h0->Z0Z0->4e so I set MSEL=16 with all h0 decay channels 'off' but mdme(207,1), all Z0 decay channels 'off' but mdme(164,1). A weird message appears: 'Warning: requested subprocess 103 has vanishing cross section. Process switched off!'. Is this message related to the fact that I impose some decays to be 'off' or 'on', is the event record reasonable?

Answer: No! you should get that message anytime you use MSEL=16. Process 103 is gamma+gamma. →h0, which is impossible in a pp collider, unless you define parton distributions for photons inside a proton, which is conceivable and might happen one day. However, till then, the main application is e+e- colliders. At onset, the program has no table of which process is allowed at what machine, but determines that dynamically at initialization. The message thus is perfectly harmless.

6

-Why the 2 generations in a pp collider: 1) h0 production with only MSEL=16 and 2) MSEL=0 plus MSUB(102, 123, 124) = 1 give exactly the same event-record?

Answer: MSEL=16 is equivalent to MSEL=0 with processes 3, 102, 103, 123 and 124 'on'. 103 is switched 'off' for an obvious reason (production of h0 by gamma gamma) so does not count. The difference therefore sits in process 3 (which could be replaced by 121 and 122 for a better kinematics of top). The event records should diverge. However it could well happen that the first few events still agree (if they do not come to involve process 3). If you also include process 3, then MSEL=16 and =0 options clearly are equivalent and should agree through the whole run.

7

-I understand that for the unphysical h0 and Z0 decay channels I should set them to '-1' rather than '0'. What is the difference between the 2 settings?

Answer: You are right, an unphysical channel should have '-1' rather than '0'. The idea is: '-1' leaves no trace for the channel behind, '0' closes the channel for decay but still allows it in the calculation of the total width of the Higgs (or whatever particle you study).

8

-How may I get the proton parton distribution CTEQ4L and CTEQ4M (PYTHIA stopped brutally at initialization)?

Answer: You should set **MSTP(51)=4032** for **CTEQ4L** or **4034** for **CTEQ4M** and **MSTP(52)=2** (choice of the **pdflib** proton parton distribution function library instead of the PYTHIA internal one). PYTHIA stopped because you (likely) forgot to link the library **pdflib** (when such a library is required and everything is OK, PYTHIA writes the taken distribution at initialization. You also have to watch out not to link the dummy **pdflib**-lookalike routines distributed with PYTHIA in order to avoid crashes when **pdflib** is not linked).

9

-To generate pp gives cc~ (bb~) is MSEL=4 (5) a good choice in despite that processes 84 and 85 are set 'off'?

Answer: Yes for both channels! The **MSEL** values allow processes of relevance both in e+e-, ep and pp collisions, and then at initialization the irrelevant ones for the current machine are switched off. However there are kinds of problems, as discussed in the manual, section 8. In particular, if you study charm production at high energies, then production in the hard subprocess, **MSEL=4**, is no longer the dominant mode of charm production. Instead you can have ordinary jet events where charm is produced in the parton shower evolution. The same, but somewhat less severe (because of the higher mass) for b.

10

-When generating pp->h0->Z0Z0->4e for a Higgs mass between 230 and 231 GeV, I often get the message: 'Warning: maximum violated by 1.021E+00 in event... ISUB=102; Point of violation: tau = 8.393E-04, y* = 3.524E-01, cthe = .240251, tau' = 1.077E-01 XSEC(102,1) increased to 8.025E-12' and the message: 'Advisory warning type 3 given after 928 LUECXEC calls: (LUROBO:) boost vector too large'. Are these serious?

Answer: Violations so close to 1 is no problem, it just meant that the initialization was not fully efficient in finding the maximum; Even somewhat bigger numbers normally are acceptable. The boost warning could be more severe. Likely it is related to the treatment of the beam remnants, where the P/E ratio of partons and hadrons can get quite close to unity. **In PYTHIA 5.7** single precision is used in most places, and this is then at the root. **With PYTHIA 6.1**, one main change is the use of double precision throughout, which reduces this kind of problem. However, most likely the

boost warning is still handled correctly, and even when it is not, the error will be associated with particles in the extreme direction, which anyway are not seen in your detector. So there is no reason to be really anxious unless this kind of warnings are very frequent.

11

-I am concerned by the multiplicity and I tried to increase the size of the event-record array by setting MSTU(4) to 10000. PYTHIA crashed! What happened?

Answer: Did you actually perform a global substitute of /LUJETS/ to the bigger dimensions in all of PYTHIA and JETSET, and recompile these programs? Only that way would addressing be properly handled. **MSTU(4)=10000** does not physically change the common block size in the code, but only tells how far the routines are allowed to write before they would go out of bounds. So if this is 10000 and the actual routine is still sized by 4000 in some routine, this routine would confuse whether it was reading from **K** or the **P** array. So the crash.

12

-What is a reasonable value of PARP(82) with the other (than the default) parton distribution functions (namely CTEQ4L)?

Answer: That is not a simple question. The value has to be taken from a fit to data e. g. the average charged multiplicity, since it is a nonperturbative parameter. You will find the current standpoint in the PYTHIA 7.1 update notes, in the UNDERLYING EVENTS section. The formulæ given there, which also imply an energy dependence, would be a decent first bet also for CTEQ4L.

IV-8) How to compile and load a program:

Assuming your machine (running under Linux) has an access to the CERN libraries, then to compile and load the program «mypythia.f» (FORTRAN interface to PYTHIA), you should define the file «Makefile» (the «make» description file) as:

```
FC=g77
OBJ=mypythia.o
LIB=`cernlib pdflib pythia6136 pawlib graflib mathlib packlib kernlib`
mypythia: $(OBJ)
$(FC) -o $@ $(OBJ) $(LIB)
```

This would compile and load (at CERN) the proper PYTHIA 6.1 libraries (with its satellites, see **FAQ # 8**).

i) to get the executable «mypythia», type «make mypythia»,

ii) to run it, type, «mypythia».

Note 1: This is an example that works at CERN (you should adapt it to your machine once the proper link to CERN libraries is performed).

Note 2: **To load the proper PYTHIA 5.7 libraries the variable LIB should be:**
LIB=`cernlib pdflib isajet744/obsolete jetset74 pawlib packlib mathlib kernlib`

IV-9) Downloading a tutorial script:

THIS PAGE IS NOT YET BUILT SORRY!

V) Acknowledgements

My most grateful thanks go to:

-**Torbjörn Sjöstrand**, one of the main author of PYTHIA, who, with patience and kindness, answered all my questions and checked this document for errors. Of course, in spite of our vigilance, if some error has crept in, it would be my responsibility and not his,

-**Denise Dufournaud**, a computer scientist of LAPP, who wrote the tutorial interface to PYTHIA, With her help, this tutorial can be run successfully on many different systems,

-**Gerard Dromby**, the Web-page designer of LAPP, without whose help this miniguide containing many mathematical formulæ would never have appeared with such clarity and elegance on the Internet.