Implementation of the Monte Carlo Event Generator SHERPA and Analysis

Summer Student Programme 2010

Steffen Henkelmann

Georg-August University of Goettingen

steffen.henkelmann@desy.de

Supervisors: Dr. G. Grindhammer, R. Kogler

H1 Group, MPI Munich

guenterg@desy.de,roman.kogler@desy.de

Abstract

During the DESY Summer School 2010, I was a participant of the H1 Group of the Max Planck Institute Munich supervised by Dr. Guenter Grindhammer and Roman Kogler. I implemented the Monte Carlo Generator SHERPA in the H1 Analysis Software H1OO and analyzed its predictions for jet production in neutral current deep-inelastic scattering. Comparisons to the predictions from the MC Generators Django and RapGap are made. This report gives a short summary of my work and endeavors to provide documentation on how to run SHERPA in the H1 Analysis Software H1OO.

Contents

1 Introduction

1.1 The H1 Experiment

The H1 Detector was one of four experiments at the lepton-proton collider HERA at DESY located in Hamburg, Germany. The HERA collider was in operation from 1992 to 2007 during which H1 collected data corresponding to an integrated luminosity of $0.5fb^{-1}$. One of the main goals of H1 is to understand and measure the structure of the proton, to study the fundamental interactions between particles and to search for physics beyond the Standard Model. One of the main research interests of the H1 group MPI Munich is the precision measurement of jet production in ep scattering. These measurements provide important tests of pertubative QCD and can be used to extract the strong coupling α_s as well as parton density functions with high precision.

1.2 deep-inelastic scattering (DIS)

High energy lepton-hadron scattering experiments had shown for the first time that nucleons have internal structure. Scattering occurs via the exchange of a virtual particle with

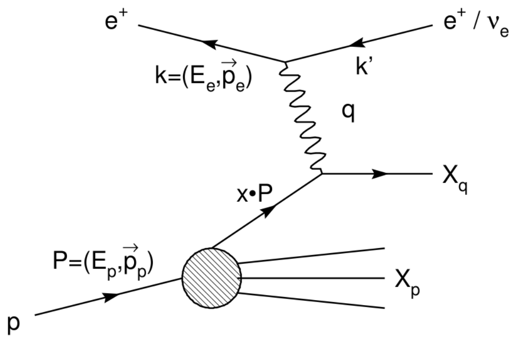


Fig. 1: Feynman Graph in the obtained DIS process (Leading order)

four-vector q, as shown in figure 1. In the case of neutral current interactions, the virtual particle is either a photon or a Z^0-boson. The wavelength of the virtual particle is inversely proportional to the squared momentum transfer $Q^2 = -q^2$ from the electron to the target. This means by increasing the momentum transfer the wavelength of the virtual particle decreases and the resolution increases. Assuming that the target particle is point-like, increasing the momentum transfer will cause all momentum to be carried by the target particle (elastic scattering, where the invariant mass of the hadronic final state M_x^2 is approximately the invariant mass of the proton m_p^2). If not, the target particle must have constituents that are carrying a part of the whole momentum (deep-inelastic scattering $Q^2 >> m_p^2$ and $M_x^2 >> m_p^2$). The fraction of the momentum of the proton carried by the struck parton in leading order is called x_{Bj}.
In the experimental setup of HERA, the center-of-mass energy $\sqrt{s}$ was fixed. During HERA RUN II the proton beam energy was 920 GeV and the lepton beam energy was 27.6 GeV. By measuring the scattered lepton we can easily calculate Q^2 by subtracting the four-vector of the incoming lepton from the scattered electron, $Q^2 = -(k - k^{'})^2$. The Bjorken scaling

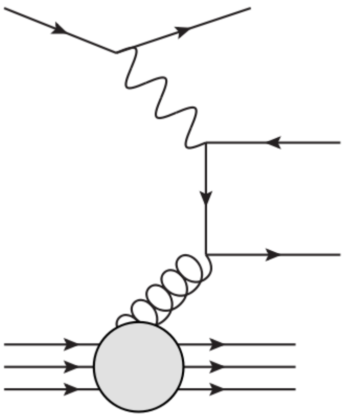
variable x_{bj} and the inelasticity y are defined via

$$y = \frac{P \cdot q}{P \cdot k} \text{ and } x = \frac{Q^2}{2 \cdot p \cdot q}, \tag{1}$$

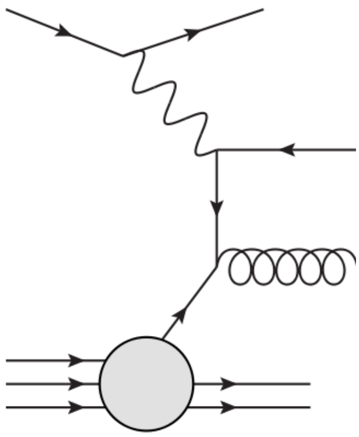
where only two variables are independent since $Q^2 = s \cdot x \cdot y$ holds. Thus, the DIS phase space is defined by specific regions in Q^2 and y which is described in section 4.7.

1.3 Multijet Production

In figure 1 the lepton-proton interaction in leading order is shown which is independent of the strong interaction. It is possible to achieve sensitivity to the strong coupling α_s by investigating multijet production. The leading order processes of 2-jet production (not containing the proton remnant jet) are boson-gluon fusion and QCD compton processes, which are shown figure 2a - 2b.



(a)



(b)

Fig. 2: Generic Feynman graphs of (a) the boson-gluon fusion process (BGF) and (b) the QCD Compton process (QCDC).

1.4 H1 Analysis Software H1OO

The H1 Analysis Software **H1OO** serves as an object-orientated analysis framework for H1 written in C++. The advantage of the software is not only the unification of existing

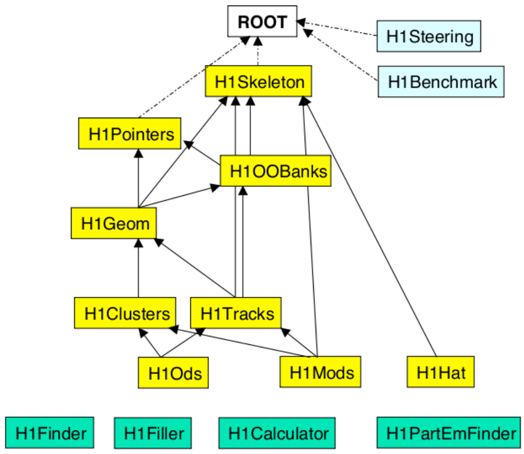


Fig. 3: Overview of the H1OO architecture

analyses algorithms, but also the unification of physics particle definitions across working groups, which provides an optimized information exchange. These points facilitate an easier handling and time saving in analysing the data.

1.5 Monte Carlo Generator

Monte Carlo Generators are based on a repeated random sampling of complicated multidimensional integrals, to compute numerical results. In the context of High Energy Physics they can be used to generate all possible configurations in the phase space. Instead of integrating over the whole phase space analytically, they generate events as random points in the phase space. Every point in the phase space is a different physicical configuration.

1.6 MC Sherpa

SHERPA is a Monte Carlo generator whose first version (SHERPA 1.1) was released at the end of 2008. SHERPA is an acronym for **S**imulation of **H**igh **E**nergy **R**eactions of **PA**rticles. It is a general-purpose tool for the simulation of particle collisions at high-energy colliders [1]. Providing a complete event generation framework written in C++, SHERPA make a large modularity available which supplies a wide range of different physics aspects. It is possible to set up more than one matrix-element generator or parton shower module at the same time. The event generation framework delivers not just plenty of possibilities for choices by the user, it also includes interfaces to commonly used input and output analysis formats like LHAPDF, HEPEVT or HEPMC. According to the authors it is easy to implement new physics models in to SHERPA.

1.7 Implementation of SHERPA in H1OO

An overview of the different steps to implement SHERPA in H1OO is given in the figure 4.

To implement SHERPA in the H1 framework it is necessary to build an interface which can communicate between the MC Generator and the Analysis software. My main task was to ensure said communication and to guarantee proper information flow between the different blocks shown in figure 4. H1GEN calls SHERPA and fills the `COMMON/HEPEVT`. With the information stored in `HEPEVT`, it is possible to fill different BOS BANKS (see below). The BOS BANKS include information about the beam momenta, polarisation, particle 4-vectors, kinematic variables, PDFs, event weights, information on the used generator modules, etc. DST is an output format which reads in the information stored in the BOS banks and writes files which can be read in by H1OO. Within H1OO the generated data files are analyzed and compared to different models with the help of the ROOT-framework.

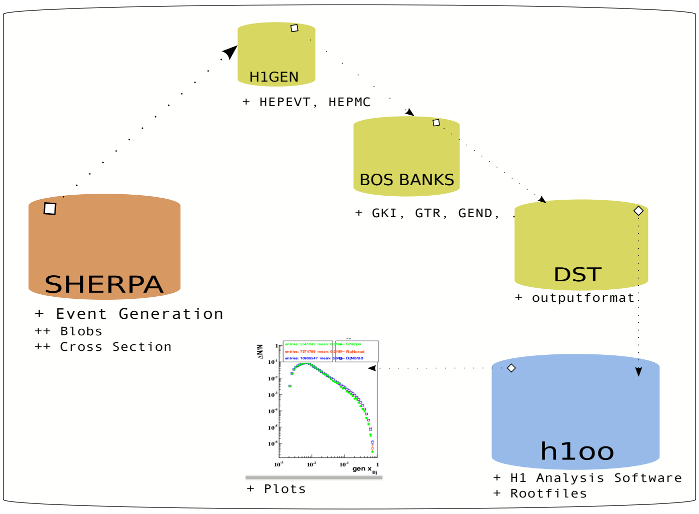
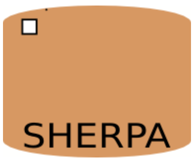


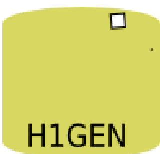
Fig. 4: `Steps to implement SHERPA`

2 SHERPA



SHERPA contains a tree-level matrix-element generator for the calculation of hard scattering processes which occur in deep-inelastic scattering (DIS). Furthermore, it is able to simulate the soft-photon radiation in the initial state (`ISR`). To get very precise predictions, the generated data need to contain nearly all phenomena which occur in a real event. Up to now SHERPA is not able to calculate QED corrections for the final state which is shown in section 4.3. To run SHERPA one needs to set up a steering file called `Run.dat`. The steering file contains information about the processes the MC Generator should create, about the beam, about the number of events, etc. An example for the `Run.dat` steering file is shown in figure 6. Further information about the configuration of the steering file `Run.dat` is provided by the documentation of SHERPA [1].

3 Implementation process



During my work we implemented SHERPA in the H1 main program called H1GEN. It was made possible to call SHERPA from the FORTRAN framework H1GEN and fill the information into `COMMON/HEPEVT`. After the filling process one is able to sort all the information one needs for further analysis. I wrote a routine which fills different information about every event in the BOS BANKS which constitute the basis for the H1 analysis software H1OO. Additionally, the routine provides information about different kinematic variables like Q^2, x and y as they are defined by the description of the BOS banks by H1. A list of the different BOS BANKS and the information they contain are given in table 1.

[1] `http://projects.hepforge.org/sherpa/doc/SHERPA-MC-1.2.1/Sherpa_2.html#SEC7`

BOS Banks	containing information	filled/not filled
HEAR	essential bank for data, for MC includes beam momenta, polarisation	✓
GHD	information on generator modules used	✓
GKI	standard and process specific kinematic variables, PDFs, weights	✓
GTR	beams, partons, undecayed and decayed particles	✓
GVX	event vertex and secondary particle vertices	✗
GEVC	event classification bank	✓
GEND	process and subprocess cross sections	✗

Tab. 1: A list of the different BOS Banks and the information they contain.

4 Analysis and Results

In order to test the implementation, I created some plots using a modified Example program called `boostedjets`. For this purpose I used tools provided by H1OO, namely H1Filler and H1Calculator. No problems with the filling of the histograms occurred. However, the efficency of SHERPA poses problems. Using the Matrix Element Generator `Comix`, the running time without weighing the events is not practical ($\sim$ 1 year). By generating events with weights, the running time is dramatically reduced, but the weights are very large (up to 10^8) and have an extremely large spread (figure 5).

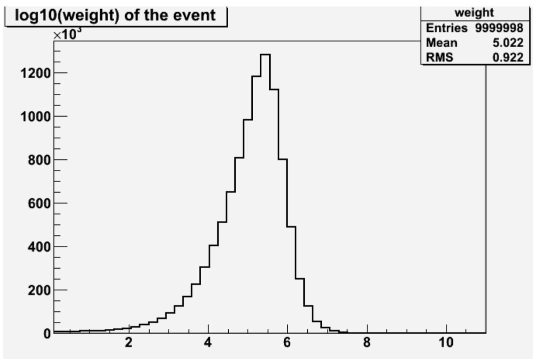


Fig. 5: Weight distribution with logarithmic values on the x-axis, for SHERPA without ISR.

Thus, up to now SHERPA can not be used on Detector Simulation Level. But it is possible to make statements about the predictions of SHERPA by comparing it to other MC Generators. For this purpose I used the analysis program `JetsAtHighQ` developed by R. Kogler. In the following plots results obtained with SHERPA in the H1 analysis framework are shown.

4.1 MC Comparison

4.1.1 Event Selection and Phase Space

For the jet analysis the phase space is restricted to $150 < Q^2 < 15000$ and $0.2 < y < 0.7$. The jet phase space is given by

- **inclusive jets**: $7 < p_t < 50$ GeV in the Breit Frame[2] and $-1 < \eta < 2.5$ in the laboratory rest frame
- **dijets**: $5 < P_t < 50$ GeV in the BF, $-1 < \eta < 2.5$ in the laboratory frame and $M_{jet_{12}} > 16$ GeV

The following descriptions and the shown plots are a selection of the plots I created and are based on inclusive jets (except the plot shown in figure 14). All plots created with the `JetAtHighQ2` can be found in the following directory:

`/h1wgs/h1mpim13/x02/usr/shenkel/Plots`

The Plots generated with `boostedjets` can be found in the rootfiles `sherpaNoISR.root` and `sherpaISR.root` in the directory

`~/shenkel/h1/H1Examples/`

4.2 Q^2 distribution

Figure 8 shows the distribution of Q^2 without (a) and with (b) initial state radiation. The distribution of SHERPA do not exactly match with the predictions given by Django and RapGap, particiularly in the range $200 < Q^2 < 600$ GeV^2. Differences in the distribution of SHERPA with and without ISR are also seen in the inelasticity y.

4.3 Jets on Hadron Level

The jet-ϕ distribution is given in figure 9. The plot (a) shows the distribution without initial state radiation. The large weights of SHERPA per event, explain the fluctuations at $-150° < \phi < 150°$). In the region $\phi > \pm 150$, SHERPA makes different predictions compared to RapGap and Django. A possible explanation is that SHERPA generates more QPM type events with additional parton shoewer(s) and that at least one jet has a large enough p_t in the BF. Plot (b) shows the same distribution with initial state radiation.In the region around $\phi = 0$ SHERPA does not show a peak. The reason for this is that SHERPA does not include final state photon radiation. For $|\phi| > 150$ the predictions made by SHERPA are different from those of the other MCs. Again, one possible reason for this is a different mixture of QPM events to the jet sample.

[2] The Breit Frame is a frame of reference in which $2 \cdot x \cdot \vec{p} + \vec{q} = 0$, and the scattered electron defines the x-axis ($\phi = 0$). The BF is used to discriminate between QPM and QCDC and BGF processes, which is achieved by requiring significant jet p_t measured in the BF.

4.4 x_{bj}- and y- distribtion

The generated x_{bj}-distribution (figure 10a) has a peak at $x_{bj} \approx 0.01$. The peak is explainable taking the definition of the phase space into account. Q^2 and y are cutted which is shown in figure 10c and figure 8 (a) and (b). Furthermore, figure 10c shows that for small values of y, the predictions of SHERPA are slightly below RapGap and Django. The information filled in the BOS BANKS (plots with gki) match with the values for the kinematic variables x and y provided by H1OO.

4.5 P_z^h-distribution

The P_z^h-distribution in figure 12 shows the generated momentum of all hadrons in z-direction (p-beam direction) in the range of $4° < \theta < 155°$. Because of the fact that the measured data lies between the distributions of Django and RapGap, we hope that the predictions of SHERPA will match better with the data than Django and RapGap. In the range between 15 and 20 GeV, the predictions of SHERPA are between the distributions of the other MCs. In the region 20 to 35 GeV the distribution of SHERPA are below the other MCs. For $P_z^h > 40$ GeV the distribution is between the other MCs. SHERPA will hopefully be able to make reasonable predictions.

4.6 Generated Jet Multiplicity

Figure 12 shows that the number of generated jets matches with Django and RapGap which is also hopeful.

4.7 Pseudorapidity η - distribution in lab frame

Figure 13 shows the distribution of the pseudorapidity η of jets in the laboratory rest frame. SHERPA's predictions do not agree with the other MCs. Its distribution is shifted slightly more forwarded, which is also suggested by the data.

4.8 Invariant Mass M_{12} in dijet events

Figure14 shows the invariant mass of the two leading jets in dijet events.

5 Summary and Outlook

I successfully implemented the Monte Carlo Generator SHERPA in the H1 Analysis framework. In addition to represent a part of providing the possibility to do further analysis with SHERPA in H1, I acquired the capability to write codes in FORTRAN and got a better understanding of ROOT and C++. I analyzed the predictions made by SHERPA for jet production in neutral current deep-inelastic scattering.

6 Acknowledgment

I really enjoyed my stay at DESY in Hamburg. I want to thank my supervisors Dr. Guenter Grindhammer and Roman Kogler who both answered all my questions and who prepared an atmosphere where it was a pleasure to work in. It was great to be a part of Dr. Grindhammers group.

I also want to say thanks to Prof. Dr. Joachim Meyer and Andrea Schrader who organized the DESY Summer Student programme. Thanks for offering such a great opportunity for students being interested in High Energy Physics.

References

[1] T. Gleisberg, S. Hoeche, F. Krauss, M. Schoenherr, S. Schumann, F. Siegert, J. Winter, "Event generation with SHERPA 1.1", JHEP 0902:007 (2009), [arXiv:0811.4622v1 [hep-ph]].

[2] Tancredi Carli, Thomas Gehrmann, Stefan Hoeche, Hadronic final states in deep-inelastic scattering with Sherpa, Eur.Phys.J.C67:73-97 (2010) [arXiv:0912.3715v1 [hep-ph]].

[3] H. Fleischhack, "Scale Dependence and Scale Uncertainties due to Missing Higher Orders in NLO Jet Calculations", DESY 2009, http://www.desy.de/f/students/2009/reports/fleischhack.pdf.

7 Appendix

7.1 SHERPA

The following sections contain information on how to setup and run SHERPA.

7.1.1 Running Sherpa

As described in section 2, to run SHERPA it is essential to set up a steering file called `Run.dat` (figure 6). The following explanation is not detailed. The SHERPA Manual[3]. provides a lot of detailed information. I am going to explain the Run.dat I used to implement

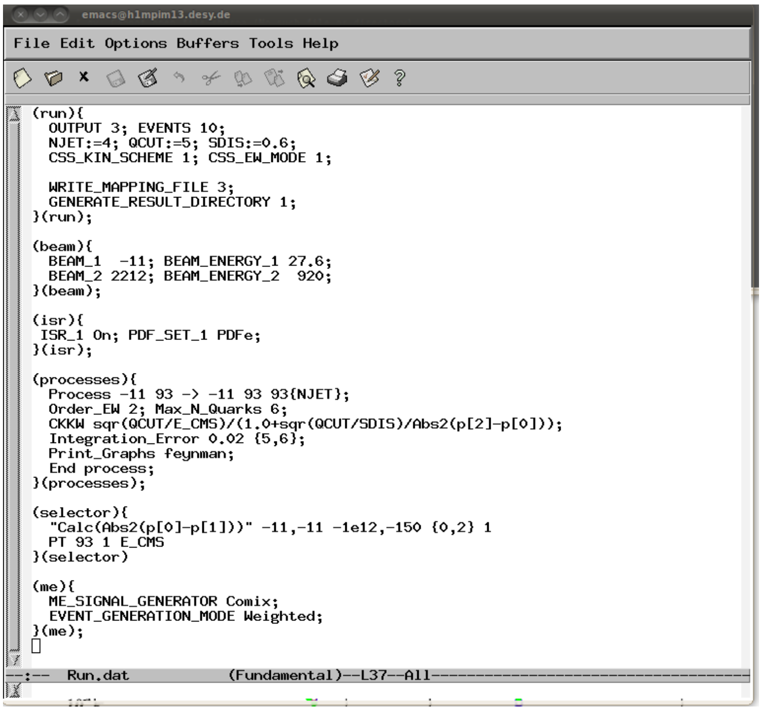


Fig. 6: `An Example for a Run.dat steering file.`

SHERPA in the H1 Framework. On the top of the steering file there is the `(run)` section which contains information about the events which will be generated.

run section

```
(run){
  OUTPUT 3; EVENTS 10;
  NJET:=3; QCUT:=5; SDIS:=0.6;
  CSS_KIN_SCHEME 1; CSS_EW_MODE 1;

  SHERPA_LDADD=SherpaHEPEVT;
  ANALYSIS=HEPEVT;
```

[3] `http://projects.hepforge.org/sherpa/doc/SHERPA-MC-1.2.1/`

```
#  ANALYSIS=HZTool,HEPEVT;

  WRITE_MAPPING_FILE 3;
}(run);
```

1. The output value can be any sum of the following:
   - 0: Error messages (it is always displayed).
   - 1: Event display.
   - 2: Informational messages during the run.
   - 4: Tracking messages (lots of output).
   - 8: Debugging messages (even more output)
2. Set the event number you want to generate (EVENTS <number of events>).
3. Set the number of maximal jets being generated (NJETS:= <number of jets>)
4. Command to fill the information provided by SHERPA to the COMMON/HEPEVT (SHERPA_LDADD=SherpaHEPEVT;
5. Decide on an analyzing method (ANALYSIS=HEPEVT;)

beam section

```
(beam){
  BEAM_1  -11; BEAM_ENERGY_1 27.6;
  BEAM_2 2212; BEAM_ENERGY_2  920;
}(beam);
```

1. Set the right beam and particles. In our case the beam energy of the e^- (-11) was 27.6 GeV and the proton (2212) beam energy was 920 GeV for HERA-2. The numbers -11 and 2212 correspond to the PDG particle code.

isr section

```
(isr){
 ISR_1 On; PDF_SET_1 PDFe;
}(isr);
```

1. Set the initial state radiation. Turn it off by commenting it with # .

processes section

```
(processes){
  Process -11 93 -> -11 93 93{NJET};
  Order_EW 2; Max_N_Quarks 6;
  CKKW sqr(QCUT/E_CMS)/(1.0+sqr(QCUT/SDIS)/Abs2(p[2]-p[0]));
  Integration_Error 0.02 {5,6};
  End process;
}(processes);
```

1. Set the right process. For further information read [2].

selector section

```
(selector){
  "Calc(Abs2(p[0]-p[1]))" -11,-11 -1e12,-60 {0,2} 1
  PT 93 1 E_CMS
}(selector)
```

1. Set the event selector. Set the minimal value of Q^2 (`-60`).

me section

```
(me){
  ME_SIGNAL_GENERATOR Comix;
  EVENT_GENERATION_MODE Weighted;
}(me);
```

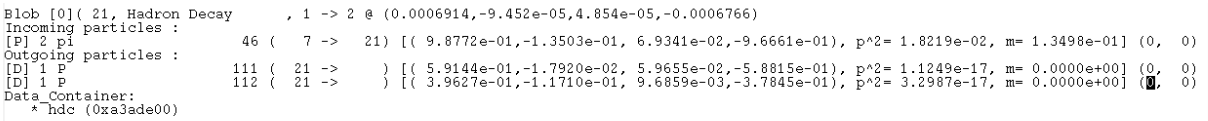
1. Set the Matrix Element Generator (e.g. `Comix, Amegic`)
2. Set `Weighted` or `Unweighted` event generation.

Set up everthing properly SHERPA should run with the command

```
~$ sherpa -g -f Run.dat > outputfile
```

7.1.2 SHERPA output

In this section I am going to explain the output of SHERPA.



```
Blob [0]( 21, Hadron Decay      , 1 -> 2 @ (0.0006914,-9.452e-05,4.854e-05,-0.0006766)
Incoming particles :
[P] 2 pi              46 (   7 ->   21) [( 9.8772e-01,-1.3503e-01, 6.9341e-02,-9.6661e-01), p^2= 1.8219e-02, m= 1.3498e-01] (0,  0)
Outgoing particles :
[D] 1 P              111 (  21 ->     ) [( 5.9144e-01,-1.7920e-02, 5.9655e-02,-5.8815e-01), p^2= 1.1249e-17, m= 0.0000e+00] (0,  0)
[D] 1 P              112 (  21 ->     ) [( 3.9627e-01,-1.1710e-01, 9.6859e-03,-3.7845e-01), p^2= 3.2987e-17, m= 0.0000e+00] (0,  0)
Data_Container:
   * hdc (0xa3ade00)
```

Fig. 7: `The output of Sherpa`

- particles have a production and a decay blob (with information of where they have been produced and where they decay) → apart from those entering the event as beams and those leaving it as final-state particles
- Blob: status [0], number (21), type (Hadron Decay), number of incoming and outgoing particles ($1 \to 3$), (t,x_1, x_2, x_3)
- Particles: particle info code, status, type, number, information about the production and decay blob, four momentum (E, p_1, p_2, p_3), the momentum- squared and mass, colour charges

The existing Monte Carlo status code convention:

0	null entry
1	final state (not decayed or fragmented)
2	decayed or fragmented
3	documentation line
4-10	reserved for future standards
11-200	used by event generators and equivalent to a null entry
201 -	available for use by detector simulation, etc.

Tab. 2: The table contains the existing Monte Carlo status code convention

7.2 Changes made to the code

Because of the fact that Sherpa was not consistent with the definition of the direction of the beam used in the H1 Framework, we needed to fix this in the interface HEPEVT_Interface.C written by Stefan Hoeche. The changes were made in the following bool function:

```
bool HEPEVT_Interface::Run(ATOOLS::Blob_List *const bl)
{
  for (size_t i(0);i<bl->size();++i) {
    for (int j(0);j<(*bl)[i]->NInP();++j) {
      Particle *cp((*bl)[i]->InParticle(j));
      if (cp->ProductionBlob()==NULL) {
Vec4D p(cp->Momentum());
p=Vec4D(p[0],p[1],-p[2],-p[3]);
cp->SetMomentum(p);
      }
    }
```

The interface of Stefan Hoeche fills the information given by SHERPA to the COMMON/HEPEVT. With this information it is possible to fill the GTR Bank.
The weight of every event is not filled in the HEPEVT but in the COMMON/HONECMN. To transmit information coming out of SHERPA to H1GEN, you can use the following environment:

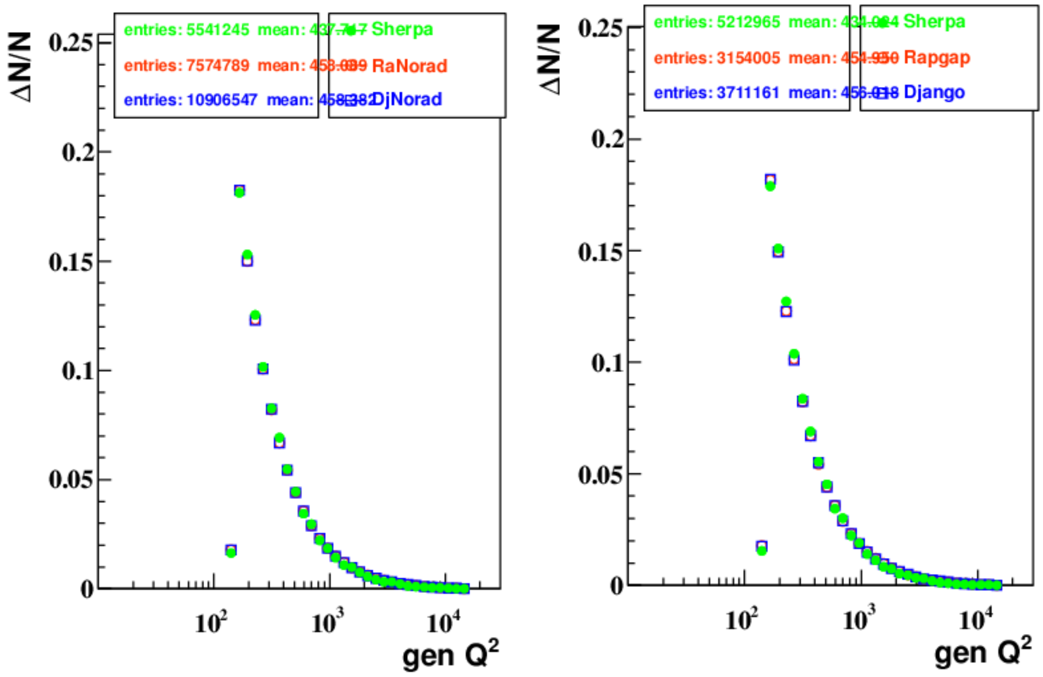
```
extern struct {
  float wtx;

} honecmn_;
```

I programmed a FORTRAN code which fills the GKI Bank. The code calculates the following:

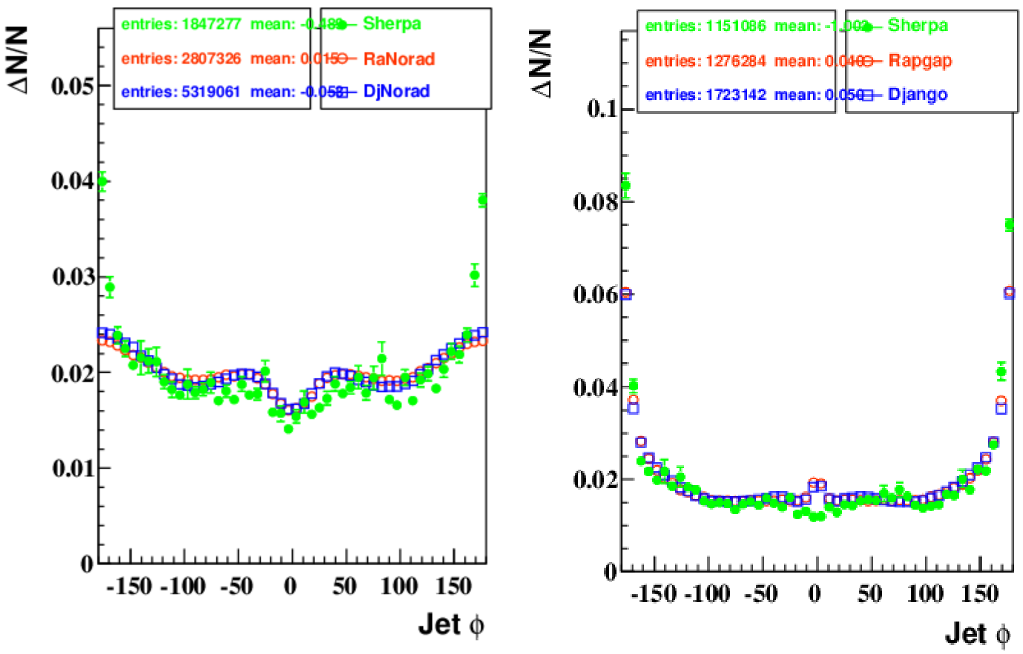
1. Initial state radiation is on or off by finding the radiated photon
2. The four vector of the scattered electron
3. The four vector of the electron which radiated a photon in the initial state
4. The four vector of the virtual photon
5. Incoming lepton beam momentum
6. Incoming proton beam momentum
7. The values of Q^2, x, y, ν.

7.3 Plots



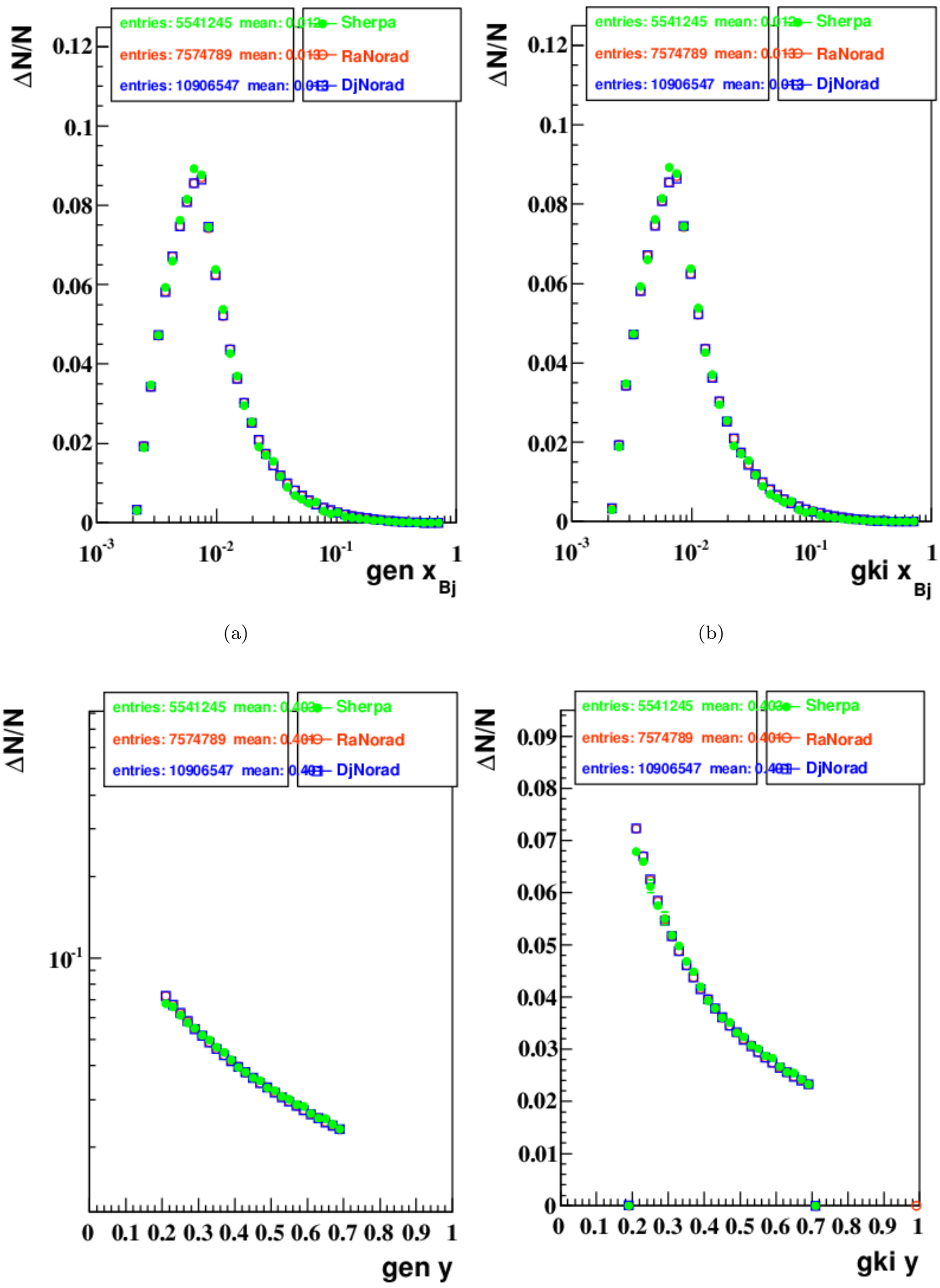
(a) (b)

Fig. 8: Q^2-distribution of generated values of Q^2 by H1OO. Plot (a) is without ISR and Plot (b) with ISR.



(a) (b)

Fig. 9: The ϕ-distribution of the inclusive Jets on Hadron Level. Plot (a) is without ISR and Plot (b) with ISR.



(a) (b) (c) (d)

Fig. 10: x_{bj}- and y- distribution with the information out of the BOS BANK gki and the generated valus by H100 (without ISR).

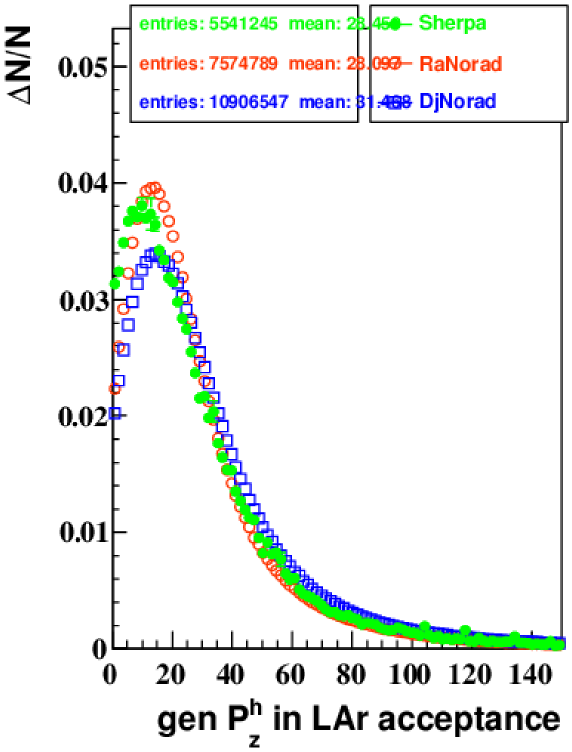


Fig. 11: `The Njet-distribution without ISR generated by H100.`

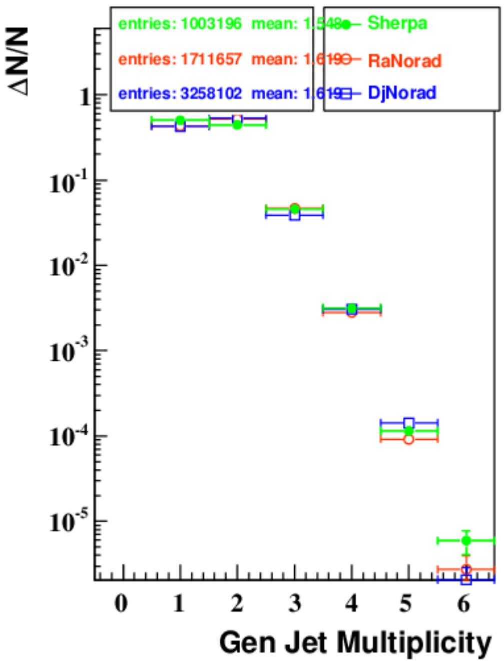


Fig. 12: `The` P_z^h`-distribution generated by H100 (without ISR).`

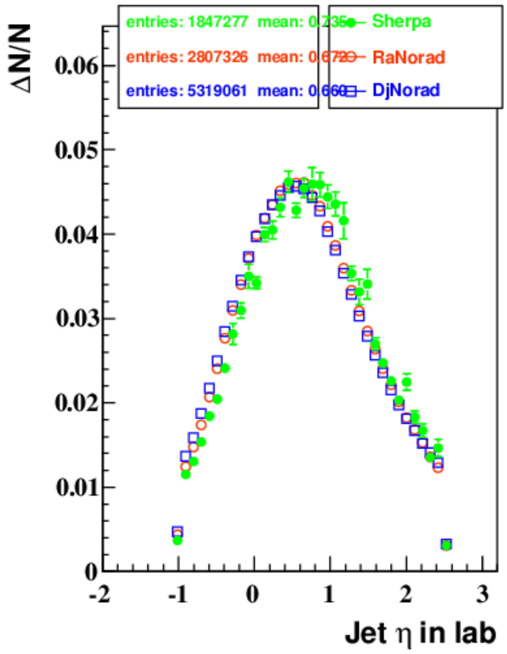


Fig. 13: The pseudorapidity η-distribution of the generated jets.

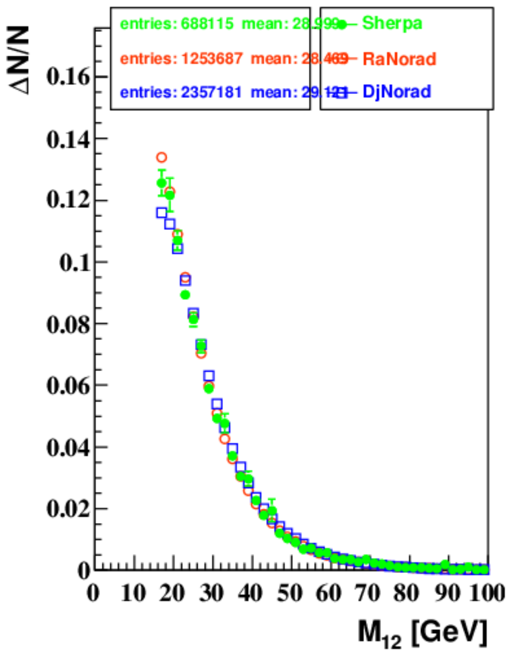


Fig. 14: M_{12}-distribution of the dijets (without ISR)