

HERA AND THE LHC

A workshop on the implications of HERA for LHC physics

March 2004 – March 2005

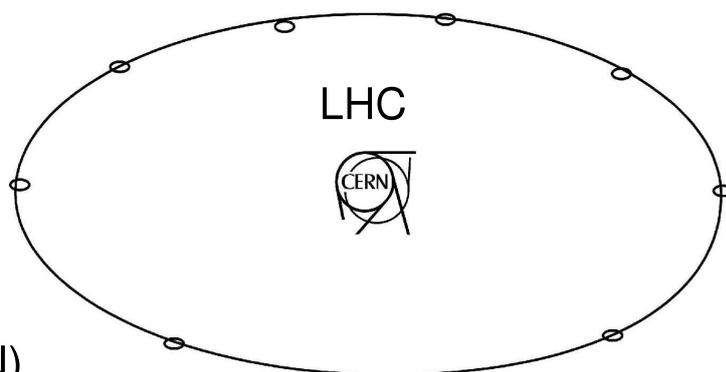
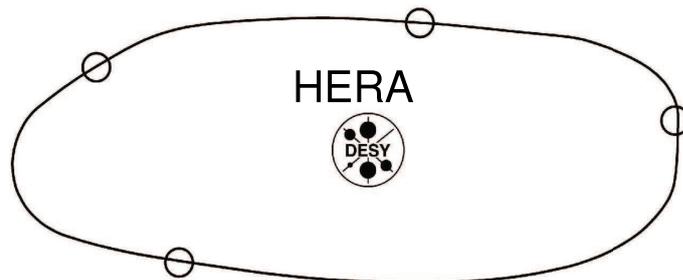
Parton density functions

Multijet final states
and energy flow

Heavy quarks

Diffraction

Monte Carlo tools



Proceedings

Editors:

A. De Roeck (CERN)

H. Jung (DESY)

Organizing Committee:

G. Altarelli (CERN), J. Blümlein (DESY), M. Botje (NIKHEF),
J. Butterworth (UCL), A. De Roeck (CERN) (chair), K. Eggert (CERN),
H. Jung (DESY) (chair), M. Mangano (CERN), A. Morsch (CERN),
P. Newman (Birmingham), G. Polesello (INFN), O. Schneider (EPFL),
R. Yoshida (ANL)

Advisory Committee:

J. Bartels (Hamburg), M. Della Negra (CERN), J. Ellis (CERN),
J. Engelen (CERN), G. Gustafson (Lund), G. Ingelman (Uppsala), P. Jenni (CERN),
R. Klanner (DESY), M. Klein (DESY), L. McLerran (BNL), T. Nakada (CERN),
D. Schlatter (CERN), F. Schrempp (DESY), J. Schukraft (CERN),
J. Stirling (Durham), W.K. Tung (Michigan State), A. Wagner (DESY),
R. Yoshida (ANL)

Contents

VI Working Group 5: Monte Carlo Tools	565
Introduction to Monte Carlo tools	567
V. Lendermann, A. Nikitenko, E. Richter-Was, P. Robbe, M. H. Seymour	
The Les Houches Accord PDFs (LHAPDF) and LHAGLUE	575
M.R. Whalley, D. Bourilkov, R.C. Group	
THEPEG: Toolkit for High Energy Physics Event Generation	582
L. Lönnblad	
PYTHIA	585
T. Sjöstrand	
HERWIG	586
M.H. Seymour	
Herwig++	588
S. Gieseke	
The event generator SHERPA	590
T. Gleisberg, S. Höche, F. Krauss, A. Schälicke, S. Schumann, J. Winter	
ARIADNE at HERA and at the LHC	592
L. Lönnblad	
The Monte Carlo event generator AcerMC and package AcerDET	596
B. Kersevan, E. Richter-Was	
RAPGAP	601
H. Jung	
CASCADE	603
H. Jung	
Leading proton production in ep and pp experiments: how well do high-energy physics Monte Carlo generators reproduce the data?	605
G. Bruni, G. Iacobucci, L. Rinaldi, M. Ruspa	

NLOLIB - A common interface for fixed-order calculations	611
K. Rabbertz, T. Schörner-Sadenius	
HZTool	617
J.M. Butterworth, H. Jung, V. Lendermann, B. M. Waugh	
CEDAR	621
A. Buckley, J.M. Butterworth, S. Butterworth, L Lönnblad, W J Stirling, M Whalley, B M Waugh	
RunMC: an object-oriented analysis framework to generate Monte Carlo events for current and future HEP experiments	624
S. Chekanov	
A C++ framework for automatic search and identification of resonances	630
S. Chekanov	

Part VI

Working Group 5: Monte Carlo Tools

List of participants in the working group

V. Lendermann, S. Nikitenko, E. Richter-Was, P. Robbe, M.Seymour

P. Bartalini, O. Behnke, D. Beneckestein, D. Bourilkov, G. Bruni, A. Buckley, A. Bunyatyan, C. Buttar, J. Butterworth, S. Butterworth, S. Caron, S. Chekanov, J. Collins, M. Corradi, B. Cox, Z. Czychula, A. Dainese, G. Davatz, R. Field, A. Geiser, S. Gieseke, T. Gleisberg, P. Golonka, G. Grindhammer, R. Craig Group, O. Gutsche, S. Höche, G. Iacobucci, S. Jadach, A. W. Jung, H. Jung, B. Kersevan, F. Krauss, S. Lausberg, N. Lavesson, K. Lohwasser, L. Lönnblad, M. Mangano, S. Maxfield, J. Monk, A.Moraes, C.Pascaud, K.Rabbertz, P.Richardson, L.Rinaldi, E.Rodrigues, M.Ruspa, A.Schälicke, T. Schörner-Sadenius, S. Schumann, T. Sjöstrand, M. Skrzypek, A. Sobol, J. Stirling, P. Szczypka, P. Thompson, N. Tuning, J. Turnau, B. Ward, Z. Was, B. Waugh, M. Whalley, J. Winter

Introduction to Monte Carlo Tools

*V. Lendermann*¹, *A. Nikitenko*², *E. Richter-Was*³, *P. Robbe*⁴, *M. H. Seymour*⁵

¹ Kirchhoff-Institut für Physik, Universität Heidelberg, Im Neuenheimer Feld 227, 69120 Heidelberg, Germany;

² Imperial College, London, UK; on leave from ITEP, Moscow, Russia;

³ Institute of Physics, JU/IFJ-PAN, 30-059 Krakow, ul. Reymonta 4, Poland;

⁴ Laboratoire de l'Accélérateur Linéaire, Université Paris-Sud, 91898 Orsay, France;

⁵ School of Physics & Astronomy, University of Manchester, UK, and Theoretical Physics Group, CERN.

Abstract

The activities of working group 5 'Monte Carlo Tools' of the HERA–LHC Workshop are summarized. The group concerned itself with the developments and tunings of Monte Carlo models in light of the HERA–LHC connection, interfaces and libraries for parton density functions, Monte Carlo running, validation and tuning frameworks as well as some data analysis tools.

1 Introduction

The goals of working group 5 were

- to review existing and developing Monte Carlo (MC) models used for studies at HERA and the LHC;
- to examine and possibly improve MC models for the LHC physics using HERA data;
- to prioritize possible measurements at HERA which would allow tuning of these MC models;
- to pursue the development of frameworks for running, validating and tuning of MC and analysis programs;
- to improve and further develop common interfaces and libraries used with MC event generators;
- to review data analysis tools developed by the HERA collaborations which can be useful for studies at the LHC.

Both theorists and experimentalists from the HERA and LHC communities came together, in order to share their experience, identify crucial issues, and discuss the future developments of the programs, libraries and frameworks.

The physics topics discussed in the group have overlapped with those of all the other working groups in this workshop. Therefore many presentations were given in common sessions with other groups, most notably with working group 2 'Multijet Final States and Energy Flows'. These presentations covered the models of multiparton interactions, new developments in parton shower models, matrix element / parton shower (ME+PS) matching and simulations of multijet final states. The contributions to the present proceedings reviewing these studies are published in the chapter of working group 2. Further contributions are summarized below.

2 Libraries for Parton Density Functions

In the past, the PDFLIB library [1] was the standard package containing parametrizations of the proton, photon and pion parton density functions (PDFs). The LHC studies have necessitated the development of a new library which should include not only the central values of PDFs but also the error sets. The PDFLIB interface appeared not well suited to meet the new requirements. Therefore a new library, LHAPDF (Les Houches Accord PDF library [2]) was created following the Les Houches meeting in

2001. During this workshop the library was extensively developed by *M. Whalley* and *D. Bourilov* [3]. Several recent PDF sets were included, both those by the leading theory groups and by the H1 and ZEUS collaborations. The fits are important for better estimations of PDF uncertainties. A particularly interesting cross check would be, for example, the comparison of the fits provided by the TeVatron collaborations with the independent predictions obtained by the DGLAP evolution of the HERA PDFs to the TeVatron region.

The photon and pion PDFs were also included in the library, thus allowing its use for all HEP analyses, in particular, for HERA studies. The library has thus developed to the level at which it can fully replace PDFLIB. Several tests of the applicability and performance of the library were made within the H1 collaboration (*V. Lendermann*).

Another topic discussed is the creation of a collection of diffractive PDF parametrizations. The project was presented by *F. P. Schilling* in a meeting of working group 4 ‘Diffraction’.

3 Monte Carlo event generators

The status and plans for major leading order (LO) and next-to-leading order (NLO) QCD programs, as well as generators with k_T factorisation were discussed.

3.1 Leading order Monte Carlo programs

Currently, the major FORTRAN MC event generators, PYTHIA [4] and HERWIG [5], are undergoing the transition to object-oriented software technologies. The C++ versions of both generators, PYTHIA7 [6] and HERWIG++ [7], are built in the common framework THEPEG [8] which is based on the CLHEP class library [9].

PYTHIA7/THEPEG includes some basic $2 \rightarrow 2$ matrix elements (ME), several built-in PDF parametrizations, remnant handling, initial- and final-state parton showers, Lund string fragmentation and particle decays. There have been plans to rework the fragmentation model, in order to include junction strings, and to implement multiple interactions (*L. Lönnblad*). However, *T. Sjöstrand* recently started a completely new C++ implementation, PYTHIA8 [10].

In parallel to the work on the C++ PYTHIA versions, the development of the FORTRAN PYTHIA6 continues. It remains the main platform for new physical concepts. Version 6.3 [11, 12] includes a completely new framework for simulation of parton showers and multiple interactions by *T. Sjöstrand* and *P. Skands*. Currently this version works for pp interactions only.

The development of HERWIG continues mainly in C++ (*S. Gieseke*, *A. Ribon*, *P. Richardson*, *M. H. Seymour*, *P. Stephens* and *B.R. Webber*). The current FORTRAN version 6.5 is foreseen as the final FORTRAN version of HERWIG (apart from possible bug fixes). It is interfaced to the JIMMY generator for multiple interactions (*J. M. Butterworth*, *J. R. Forshaw*, *M. H. Seymour*, and *R. Walker*).

HERWIG++ includes a new parton shower algorithm and an improved model of cluster fragmentation. The e^+e^- event generation is implemented in HERWIG++ 1.0. The next version including hadronic interactions is in progress. The plans for the near future are to fully implement the matrix element–parton shower matching according to the Catani–Krauss–Kuhn–Webber (CKKW) scheme [13], as well as multiple interactions. A new framework for accessing particle data and simulations of particle decays is currently being constructed by *P. Richardson*. The treatment of hadronic decays will include spin correlations.

Further physics models can be incorporated into the same THEPEG framework. In particular, it is planned to make a C++ version of ARIADNE [14] based on THEPEG (*L. Lönnblad*). ARIADNE implements the Dipole Cascade Model (DCM) [15, 16] as an alternative to the DGLAP-based shower models used in PYTHIA or HERWIG.

Despite the great success of ARIADNE in modelling hadronic final state observables, as measured at LEP and HERA, additional work is necessary to make ARIADNE fully suitable for modelling inter-

actions at the LHC. This was, in particular, shown in a study presented by *Z. Czyczula* on the impact of parton shower models on the generation of bbH , $H \rightarrow \tau\tau$ at the LHC [17]. The planned features include a remodelling of initial-state $g \rightarrow q\bar{q}$ splittings as well as the introduction of the $q \rightarrow g^*q$ process.

Another study was presented by *N. Lavesson* of ME+PS matching in ARIADNE on the example of W +jet production at the TeVatron [18]. It is also planned to include the matching to fixed order tree-level matrix elements *à la* CKKW [13, 18, 19] for the most common subprocesses at the LHC. When these plans are realised (we hope during 2006), it should be safe to use ARIADNE for LHC predictions.

An alternative C++ implementation is performed in the SHERPA program (*T. Gleisberg, S. Höche, F. Krauss, A. Schälicke, S. Schumann, J. Winter*) [20] which is capable of simulating lepton–lepton, lepton–photon, photon–photon and fully hadronic collisions, such as proton–proton reactions. In its current version SHERPA includes the ME generator AMEGIC++ providing the matrix elements for hard processes and decays in the SM, MSSM and the ADD model, the parton shower module APACIC++ containing virtuality-ordered initial- and final-state parton showers, ME+PS matching using the CKKW algorithm, the AMISIC++ module for a simple hard underlying event model taken from PYTHIA and an interface to the PYTHIA string fragmentation and hadron decays. Studies were presented on ME+PS matching considering W/Z +jet production at the TeVatron and at the LHC (*S. Schumann*) [18], and on the underlying event simulations (*S. Höche*) [12].

None of the above C++ programs is available for ep interactions yet, and so no applications and tests of these programs at HERA have been possible.

Further talks were given on the RAPGAP event generator [21] by *H. Jung* and on the ACERMC event generator [22] by *B. Kersevan* and *E. Richter-Was*. RAPGAP is one of the major generators used at HERA. It includes leading-order QCD matrix elements, LEPTO [23] and ARIADNE parton cascade models, as well as simulations of hard diffraction. Both ep and pp versions are available. Recently, the Les Houches Accord interface for fragmentation models was included; this allows the choice between the PYTHIA and HERWIG fragmentation models. This feature may allow better estimations of measurement uncertainties accounting for the transition from the parton to the hadron level of final states. It is planned to include double-diffractive scattering for pp collisions to allow simulation of diffractive Higgs production.

The ACERMC event generator simulates the Standard Model backgrounds to the Higgs production in pp collisions. It includes LO QCD matrix elements produced by MADGRAPH/HELAS [24], as well as both PYTHIA and HERWIG parton shower and fragmentation models via the Les Houches Accord interface. During this workshop, the ARIADNE parton shower model and the LHAPDF library were implemented. The program is also interfaced to TAUOLA [25] for precise treatment of τ decays and to PHOTOS [25, 26] for simulations of QED radiative decays. The study on the impact of parton shower models on generation of bbH , $H \rightarrow \tau\tau$ at the LHC [17], mentioned above, was performed using ACERMC.

The program can be linked with the ACERDET package [27] which provides a fast and simplified simulation of the expected ATLAS detector effects (energy smearing, acceptance corrections) as well as the usual analysis steps (jet reconstruction algorithms, isolation criteria, etc.). This allows a quick estimation of the feasibility of measurements in an LHC experiment, not necessarily by the members of the experimental collaboration.

3.2 NLO Monte Carlo programs

NLO QCD calculations are required to make theoretical predictions at the level of precision currently reached in particle scattering experiments. However, writing a hadron level MC program implementing an NLO model is a very complicated task, which has currently been solved only for a few pp reactions [28]. An important step forward would be an ep version of MC@NLO. It would be a major benefit for HERA studies of heavy quark and multijet production and would also allow an extensive validation of the NLO QCD calculations with HERA data. The development of the program started recently [29].

3.3 Monte Carlo programs with k_T factorization

The CASCADE event generator presented by *H. Jung* [30] provides an implementation of the CCFM model for parton cascades [31]. The program was very successful in describing hadronic final states at low x at HERA. First applications of CASCADE for the studies at the LHC were presented by *G. Davatz* [17]. The plans include an implementation of quark lines into CCFM cascades (currently, only gluon lines are implemented), as well as a new model for multiparton interactions based on the AGK cutting rules [32].

A reformulation of the CCFM model into the link dipole chain (LDC) model [33] provides a simplified formalism, which has been incorporated into the LDCMC program by *H. Kharraziha* and *L. Lönnblad* [14, 34]. An LDCMC version for deep inelastic ep scattering is available within the framework of the ARIADNE event generator. A pp version is planned.

In conjunction with these models, special sessions of working group 2 were dedicated to possibilities of determining the unintegrated parton distributions in the proton [35].

4 Comparisons of MC models with data

Models for particular subprocesses and their tuning using HERA data were reviewed in the corresponding working groups. As mentioned above, many discussions were carried out in the common sessions of working group 5 with the other groups.

One topic considered in WG5 is a comparison of leading proton data with several MC models (*G. Bruni, G. Iacobucci, L. Rinaldi, M. Ruspa*) [36]. The ep data from ZEUS and pp data from ISR and fixed-target experiments were confronted to the HERWIG (together with POMWIG [37] and SANG to simulate diffraction), LEPTO, ARIADNE and PYTHIA simulations. This exercise revealed that the simulation of the leading-proton momenta, both longitudinal and transverse to the beams, does not reproduce the properties of the data.

This study can be especially important for the understanding of diffractive processes and backgrounds for them at the LHC.

5 MC running, tuning and validation frameworks

During this workshop much progress was made in developing common frameworks that provide a convenient handling of MC and analytical programs and allow quick comparisons of MC simulations and analytical calculations with the results of HERA and other HEP experiments. The developments of HZTOOL/JETWEB, RUNMC and NLOLIB packages were presented and actively discussed.

The HZTOOL [38, 39] library provides a comprehensive collection of FORTRAN routines to produce various distributions using Monte Carlo event generators. The routines allow easy reproduction of the experimental distributions by modelling programs and give access to published data from the EMC, SPS, LEP, HERA and TeVatron experiments. A number of studies for the LHC and the future linear collider are also included. The library can be linked with all major FORTRAN MC event generators, and with a number of NLO QCD programs from the NLOLIB package (see below). The development of the library started within the workshop ‘Future Physics at HERA’ and steadily continued in the last few years.

In the current workshop, the emphasis was put on the HERA results relevant for the LHC [38]. Several measurements by H1 and ZEUS were implemented which allow tuning of multiparton interaction models in MC event generators (work by *D. Beneckenstein, A. Bunyatyan, J. M. Butterworth, H. Jung, S. Lausberg, K. Lohwasser, V. Lendermann, B. M. Waugh*). Common tunings of multiple interaction models based on the TeVatron and HERA results may constitute in the future an interesting outcome of the current efforts. Recent H1 and ZEUS results on heavy quark production in ep collisions were also added (*A.W. Jung, A. Geiser, O. Gutsche, P. Thompson*). In addition, calculations of benchmark cross-sections for heavy flavour production were included [40].

Based on HZTOOL, JETWEB [38, 41] is a facility for tuning and validating Monte Carlo models through a World Wide Web interface. A relational database of reaction data and predictions from different models is accessed through a Java servlet, enabling a user to find out how well a given model agrees with a range of data and to request predictions using parameter sets that are not yet in the database.

The transition of experimental analysis frameworks and Monte Carlo generators to object-oriented software technologies necessitates a proper development of the MC running, tuning and validation frameworks. For this reason, HZTOOL/JETWEB is currently subject to a major redesign within the CEDAR ('Combined e-Science Data Analysis Resource for high energy physics') project [42]. CEDAR should comprise:

- an extensive archive of data from particle scattering experiments, based on the Durham HEP database [43];
- validation and tuning of Monte Carlo programs, parton distribution functions and other high-energy physics calculation programs, building on JETWEB;
- access to well-defined versions of these programs and code management support for developers;
- a standardized set of data formats for specifying HEP measurements as used in HepData and Monte Carlo event generator configurations as used in JETWEB;
- Grid compatibility for distribution of JETWEB Monte Carlo submissions and to enable secure addition of experimental data to the HepData catalogue by experimental collaborations.

A particularly important step in building CEDAR will be designing a C++ equivalent for HZTOOL, as well as providing an interface to the new C++ MC event generators.

A complementary approach using the object-oriented software design was realised in the RUNMC framework [44] by *S. Chekanov*. While JETWEB is a Web server system, RUNMC is a desktop application written in C++ and Java. It provides a unified approach to generate MC events and to analyse different MC models. All major FORTRAN MC event generators can be run via RUNMC. The output of FORTRAN MC programs is converted to C++ classes for further analysis and for graphical representation (histograms). The graphical user interface of RUNMC allows an initialization of MC models and histograms in a unified manner, and provides monitoring of the event generation. The program provides an interface to HZTOOL. It also allows loading of 'project files' which can contain external calculations, MC tunings, histogram definitions, etc. In particular, these files can include C++ data analysis code, similar to the HZTOOL FORTRAN analysis routines.

A further project, discussed in working group 5, is a common framework for the NLO QCD programs, NLOLIB [45], which was initiated within the workshop 'Monte Carlo Generators for HERA Physics'. Since hadron level Monte Carlo programs implementing QCD NLO calculations are not (yet) available for many processes, parton level NLO calculations are extensively used. NLOLIB is aimed at becoming a container for virtually all NLO QCD programs. It provides:

- a set-up for compiling and linking the programs on diverse UNIX platforms;
- a unified access to the NLO event records;
- a unified steering for parameters and settings;
- a unified access to PDF libraries;
- an interface to HZTOOL, thus allowing easy comparisons with experimental results;
- examples of the analysis code which can be linked with the library.

During the workshop the structure of the framework was further developed by *K. Rabbertz*. In addition to already implemented programs for ep (DISENT [46], DISASTER++ [47], MEPJET [48]) and e^+e^- (RACOONWW [49]) physics, an effort was made to integrate further ep programs: NLOJET++ [50] (*K. Rabbertz*) and JETVIP [51, 52] (*T. Schörner-Sadenius*). The integration of the NLO programs for pp physics is surely possible, but requires additional effort.

6 Data analysis tools

Apart from MC related topics, general analysis tools, aimed at searches for specific final states, were presented.

One such tool is SBUMPS [53], currently being developed by *S. Chekanov*, which performs automatic searches for resonance peaks in invariant-mass distributions of two or more tracks. The program can be useful for searches of new states as well as for the reconstruction of known resonances.

Recently, interest in hadron spectroscopy at HERA increased sharply with the observations of narrow peaks in inclusive invariant-mass distributions which can be interpreted as pentaquarks [54]. These studies have inspired the development of the automated peak searching tool, which can be used in data analyses at any particle scattering experiment.

A general strategy for searches for new physics was presented by *S. Caron*. The approach was developed and used by the H1 Collaboration for searches of new phenomena at HERA [55]. It involves a statistical algorithm to search for deviations from the Standard Model in the distributions of the scalar sum of transverse momenta or invariant mass of final-state particles and to quantify their significance.

7 Conclusions

A number of interesting developments of MC models, programs, libraries and frameworks were presented in working group 5. The general status and prospects for major established and currently developed MC generators were reviewed. In common sessions with working group 2, the models of multiparton interactions, new developments in parton shower models, matrix element/parton shower matching and simulations of multijet final states were extensively discussed. Direct communication between theoreticians and experimentalists from the HERA and LHC communities allowed the pursuit of several developments and studies. In particular the recent advances in the development of the LHAPDF library, HZTOOL, RUNMC and NLOLIB frameworks were inspired by discussions within working group 5. It is hoped that this will help further studies on validation and tuning of the MC models for multiparton interactions, parton showers, and heavy flavour production.

References

- [1] H. Plathow-Besch, *Comput. Phys. Commun.* **75**, 396 (1993).
- [2] W. Giele *et al.*, Workshop on Physics at TeV Colliders, Les Houches, France, May 2001.
- [3] M. Whalley, D. Bourilkov and R. C. Group, these proceedings, working group 5; LHAPDF Web page, available on <http://hepforge.cedar.ac.uk/lhapdf/>.
- [4] T. Sjöstrand, P. Edén, C. Friberg, L. Lönnblad, G. Miu, S. Mrenna and E. Norrbin, *Comput. Phys. Commun.* **135**, 238 (2001); PYTHIA Web page, available on <http://www.thep.lu.se/~torbjorn/Pythia.html>.
- [5] G. Corcella, I. G. Knowles, G. Marchesini, S. Moretti, K. Odagiri, P. Richardson, M. H. Seymour and B. R. Webber, *JHEP* **0101**, 010 (2001); HERWIG Web page, available on <http://hepwww.rl.ac.uk/theory/seymour/herwig/>.
- [6] PYTHIA7 Web page, available on <http://www.thep.lu.se/Pythia7/>.
- [7] S. Gieseke, these proceedings, working group 5; HERWIG++ Web page, available on <http://www.hep.phy.cam.ac.uk/theory/Herwig++/>.
- [8] L. Lönnblad, these proceedings, working group 5; ThePEG Web page, available on <http://www.thep.lu.se/ThePEG/>.
- [9] CLHEP Web page, available on <http://wwwasd.web.cern.ch/wwwasd/lhc++/clhep/>.
- [10] T. Sjöstrand, these proceedings, working group 5.
- [11] T. Sjöstrand, L. Lönnblad, S. Mrenna and P. Skands, *Pythia 6.3: Physics and Manual*. Preprint hep-ph/0308153, 2003.

- [12] C. Buttar *et al.*, these proceedings, working group 2.
- [13] S. Catani, F. Krauss, R. Kuhn and B. R. Webber, JHEP **11**, 063 (2001);
F. Krauss, JHEP **08**, 015 (2002).
- [14] L. Lönnblad, these proceedings, working group 5;
L. Lönnblad, Comput. Phys. Commun. **71**, 15 (1992).
- [15] G. Gustafson, Phys. Lett. B **175**, 453 (1986).
- [16] G. Gustafson and U. Pettersson, Nucl. Phys. B **306**, 746 (1988).
- [17] Z. Czyczula, G. Davatz, A. Nikitenko, E. Richter-Was, E. Rodrigues and N. Tuning, these proceedings, working group 2.
- [18] S. Höche *et al.*, these proceedings, working group 2.
- [19] L. Lönnblad, JHEP **05**, 046 (2002);
N. Lavesson and L. Lönnblad, JHEP **07**, 054 (2005).
- [20] T. Gleisberg, S. Höche, F. Krauss, A. Schälicke, S. Schumann and J. Winter, JHEP **402**, 056 (2004);
T. Gleisberg, S. Höche, F. Krauss, A. Schälicke, S. Schumann and J. Winter, these proceedings, working group 5;
SHERPA Web page, available on <http://www.physik.tu-dresden.de/~krauss/hep/>.
- [21] H. Jung, these proceedings, working group 5;
H. Jung, Comput. Phys. Commun. **86**, 147 (1995);
RapGap Web page, available on <http://www.desy.de/~jung/rapgap/>.
- [22] B. Kersevan and E. Richter-Was, these proceedings, working group 5;
AcerMC Web page, available on <http://borut.home.cern.ch/borut/>.
- [23] G. Ingelman, A. Edin and J. Rathsman, Comput. Phys. Commun. **101**, 108 (1997);
Lepto Web page, available on <http://www3.tsl.uu.se/thepl/lepto/>.
- [24] F. Maltoni and T. Stelzer, JHEP **02**, 027 (2003);
MadGraph Web page, available on <http://madgraph.hep.uiuc.edu/>.
- [25] Tauola and Photos Web page, available on
<http://wasm.home.cern.ch/wasm/goodies.html>.
- [26] P. Golonka and Z. Was, these proceedings, working group 2.
- [27] AcerDET Web page, available on
<http://erichter.home.cern.ch/erichter/AcerDET.html>.
- [28] S. Frixione and B. R. Webber, JHEP **0206**, 029 (2002);
S. Frixione, P. Nason and B. R. Webber, JHEP **0308**, 007 (2003);
MC@NLO Web page, available on
<http://www.hep.phy.cam.ac.uk/theory/webber/MCatNLO/>.
- [29] S. Frixione, H. Jung and T. Toll, private communication.
- [30] H. Jung and G. P. Salam, Eur. Phys. J. C **19**, 351 (2001);
H. Jung, these proceedings, working group 5;
CASCADE Web page, available on <http://www.desy.de/~jung/cascade/>.
- [31] M. Ciafaloni, Nucl. Phys. B **296**, 49 (1988);
S. Catani, F. Fiorani and G. Marchesini, Phys. Lett. B **234**, 339 (1990);
S. Catani, F. Fiorani and G. Marchesini, Nucl. Phys. B **336**, 18 (1990);
G. Marchesini, Nucl. Phys. B **445**, 49 (1995).
- [32] J. Bartels, M. Salvadore and G. P. Vacca, Eur. Phys. J. C **42**, 53 (2005). And references therein.
- [33] Bo Andersson, G. Gustafson and J. Samuelsson, Nucl. Phys. B **463**, 217 (1996).
- [34] H. Kharraziha and L. Lönnblad, Comput. Phys. Commun. **123**, 153 (1999)

- [35] J. Collins, M. Diehl, H. Jung, L. Lönnblad, M. Lublinsky and T. Teubner, these proceedings, working group 2.
- [36] G. Bruni, G. Iacobucci, L. Rinaldi and M. Ruspa, these proceedings, working group 5.
- [37] B. E. Cox and J. R. Forshaw, *Comput. Phys. Commun.* **144**, 104 (2002);
POMWIG Web page, available on <http://www.pomwig.com/>.
- [38] J. Butterworth, H. Jung, V. Lendermann and B. Waugh, these proceedings, working group 5.
- [39] HZTool Web page, available on <http://www.cedar.ac.uk/heptools/hztool/>.
- [40] O. Behnke, M. Cacciari, M. Corradi, A. Dainese, H. Jung, E. Laenen, I. Schienbein and H. Spiesberger, these proceedings, working group 3.
- [41] JetWeb Web page, available on <http://www.cedar.ac.uk/jetweb/>.
- [42] A. Buckley, J. M. Butterworth, S. Butterworth, L. Lönnblad, W. J. Stirling, M. Whalley and B. M. Waugh, these proceedings, working group 5;
CEDAR Web page, available on <http://www.cedar.ac.uk/>.
- [43] Durham HEP database Web page, available on <http://durpdg.dur.ac.uk/hepdata/>.
- [44] S. Chekanov, these proceedings, working group 5;
RunMC Web page, available on <http://hepforge.cedar.ac.uk/runmc/>.
- [45] K. Rabbertz and T. Schörner-Sadenius, these proceedings, working group 5;
NLOLIB Web page, available on <http://www.desy.de/~nlolib/>.
- [46] S. Catani and M. H. Seymour, *Nucl. Phys. B* **485**, 291 (1997);
M. H. Seymour, DISSENT Web page, available on
<http://hepwww.rl.ac.uk/theory/seymour/nlo/>.
- [47] D. Graudenz. Preprint hep-ph/9710244, 1997;
DISASTER Web page, available on
<http://graudenz.home.cern.ch/graudenz/disaster.html>.
- [48] E. Mirkes and D. Zeppenfeld, *Phys. Lett. B* **380**, 205 (1996);
E. Mirkes and D. Zeppenfeld, *Phys. Rev. Lett.* **78**, 428 (1997).
- [49] A. Denner, S. Dittmaier, M. Roth and D. Wackerroth, *Phys. Lett. B* **475**, 127 (2000);
RacoonWW Web page, available on <http://ltpth.web.psi.ch/racoonww/racoonww.html>.
- [50] Z. Nagy, *Phys. Rev. Lett.* **88**, 122003 (2002);
NLOJET++ Web page, available on <http://www.cpt.dur.ac.uk/~nagy/nlo++/>.
- [51] B. Pötter, *Comput. Phys. Commun.* **133**, 105 (2000).
- [52] B. Pötter, *Comput. Phys. Commun.* **119**, 45 (1999);
B. Pötter, *Eur. Phys. J. Direct C* **5**, 1 (1999);
G. Kramer and B. Pötter, *Eur. Phys. J. C* **5**, 665 (1998);
M. Klasen, G. Kramer and B. Pötter, *Eur. Phys. J. C* **1**, 261 (1998);
B. Pötter, *Nucl. Phys. B* **559**, 323 (1999);
B. Pötter, *Nucl. Phys. B* **540**, 382 (1999);
JetVip Web page, available on <http://www.desy.de/~poetter/jetvip.html>.
- [53] S. Chekanov, these proceedings, working group 5;
SBumps Web page, available on <http://www.desy.de/~chekanov/sbumps/>.
- [54] A. Airapetian *et al.* [HERMES Collaboration], *Phys. Lett. B* **585**, 213 (2004);
S. Chekanov *et al.* [ZEUS Collaboration], *Phys. Lett. B* **591**, 7 (2004);
A. Aktas *et al.* [H1 Collaboration], *Phys. Lett. B* **588**, 17 (2004);
S. Chekanov *et al.* [ZEUS Collaboration], *Eur. Phys. J. C* **38**, 29 (2004);
S. Chekanov *et al.* [ZEUS Collaboration], *Phys. Lett. B* **610**, 212 (2005).
- [55] A. Aktas *et al.* [H1 Collaboration], *Phys. Lett. B* **602**, 14 (2004).

The Les Houches Accord PDFs (LHAPDF) and LHAGLUE

M R Whalley[†], D Bourilkov[‡] and R C Group[‡]

[†]University of Durham, Durham, DH1 3LE, UK

[‡]University of Florida, Gainesville, FL 32611, USA

Abstract

We describe the development of the LHAPDF library from its initial implementation following the Les Houches meeting in 2001 to its present state as a functional replacement for PDFLIB. Brief details are given of how to install and use the library together with the PDF sets available. We also describe LHAGLUE, an add-on PDFLIB look-a-like interface to LHAPDF, which facilitates using LHAPDF with existing Monte Carlo generators such as PYTHIA and HERWIG.

1 LHAPDF – Introduction

Parton Density Functions (PDFs), which describe the partonic content of hadrons, need to be well understood and of sufficiently high precision if theoretical predictions are to match the experimental accuracies expected from future LHC data. These PDFs, which are produced by several different groups (e.g. MRST, CTEQ, Alekhin and more recently ZEUS and H1), are derived from fitting deep inelastic and related hard scattering data using parameterisations at low Q_0^2 ($\approx 1-7$ (GeV/c)²) and evolving these to higher Q^2 . These PDFs are typically presented as grids in $x-Q^2$ with suitable interpolation codes provided by the PDF authors. The CERN PDFLIB library [1] has to date provided a widely used standard FORTRAN interface to these PDFs with the interpolation grids built into the PDFLIB code itself. However, it is realised that PDFLIB would be increasingly unable to meet the needs of the new generation of PDFs which often involve large numbers of sets ($\approx 20-40$) describing the uncertainties on the individual partons from variations in the fitted parameters. As a consequence of this, at the Les Houches meeting in 2001 [2], the beginnings of a new interface were conceived — the so-call “Les Houches Accord PDF” — LHAPDF. This has further been developed over the course of the HERA-LHC workshop incorporating many new features to enable it to replace PDFLIB as the standard tool to use. The development is briefly described in this writeup together with LHAGLUE, an interface to LHAPDF, which provides PDF access using almost identical calling routines as PDFLIB.

2 LHAPDF – Development during the Workshop

In its initial incarnation (Version 1), LHAPDF had two important features which distinguished it from the methods used by PDFLIB in handling PDFs.

Firstly the PDFs are defined by the analytical formulae used in the original fitting procedures, with external files of parameters, which describe the momentum x distributions of the partons at the relevant Q_0^2 . Evolution codes within LHAPDF then produce the PDF at any desired Q^2 at the users request. At present LHAPDF provides access to two evolution codes, EVLCTEQ for the CTEQ distributions and QCDDNUM 16.12 [3] for the other PDF sets. This represents a radical difference from the existing methods used by the PDF authors to present their distributions where large grid files and interpolation routines are the norm. In PDFLIB these interpolation codes and grids are essentially compiled into a single FORTRAN library. The advantage of the LHAPDF method is that the compiled code is separate from the parameter files, which are typically small. Thus to add new PDF sets does not necessarily need the code to be recompiled and the library rebuilt.

Secondly, the concept is introduced of a “set” being a related collection of PDFs (e.g. an error set) all of which are accessible to the program after initialisation of that set. This allows LHAPDF to

handle the multi-set “error” PDFs produced in recent years which give predicted uncertainties to the PDF values. All the PDFs in a set are initialised together and are therefore available to the user.

V1 was written by Walter Giele of Fermilab who in 2002 released a working version which could be downloaded from a web-site together with the parameter files for a limited number of PDF sets. There was also a manual and example files. One of the present authors MRW became involved and took over maintenance and development of LHAPDF in March 2003. The limitations of the idealised situation in V1 with respect to making LHAPDF a replacement tool for PDFLIB soon became apparent.

The primary problem was that V1 contained only a limited number of PDF sets and, since the method was reliant on the x parameterisations at Q_0^2 being available, it would be virtually impossible to include many of the older sets which are still needed for comparisons. A second and serious problem is the compute time taken in the initialisation phase of the individual members of a PDF set (i.e. calling the routine InitPDF described later). This can take in the region of 2 seconds per call on a 1GHz machine and is therefore unacceptable in the situation of a program which makes repeated use of the different members.¹

A solution introduced in LHAPDF Version 2, which helps to solve the above problems, was to include the option to make the original grid files and interpolation codes available in LHAPDF **in addition** to the V1 method of parameter files and “on-the-fly” evolution. For some PDF sets both methods would be available and for others only the latter. The operation of the program was made identical for both methods with the content of the input file (with extension “.LHpdf” for the former and “.LHgrid” for the latter) dictating which is used. Not only does this allow all the older PDF sets to be included but also there is no time penalty in changing between members of the same set since all are loaded in the initialisation phase. LHAPDF V2 was released in March 2004 including many of the older PDF sets as well as some new ones.

LHAPDF Version 3 was released in September 2004 and, as well incorporating more older and some new PDF sets (e.g. ZEUS and H1), it also included the code for LHAGLUE, a newly developing add-on interface to LHAPDF which provides PDFLIB look-alike access. In addition to having subroutine calls identical to those in PDFLIB it also incorporates a PDF numbering scheme to simplify usage. It should be noted however that, because of the greatly increased number of new PDF sets, it was not possible to follow the original numbering scheme of PDFLIB and a new one was devised. This is described in more detail in Section 5.

The major feature of Version 4, which was released in March 2005, was the incorporation of the photon and pion PDFs. All the photon and pion PDFs that were implemented in PDFLIB were put into LHAPDF using identical code and using the “.LHgrid” method. The LHAGLUE numbering scheme in these cases more closely resembles that of PDFLIB than it does for the protons.

In addition in V4 there were new proton PDFs (MRST2004 and an updated Alekhin’s a02m), a new simpler file structure with all the source files being in a single “src” directory, some code changes to incorporate access to $\Lambda_{4/5}^{QCD}$ and a more rigorous implementation of the α_s evolution as being exactly that used by the PDF author.

All the LHAPDF and LHAGLUE data and code, in addition to being made available on the new web site (<http://hepforge.cedar.ac.uk/lhapdf/>), is also included in the GENSER subproject of the LHC Computing Grid.

3 LHAPDF – Development after the Workshop

Since the last HERA-LHC meeting there has been one minor release of LHAPDF (Version 4.1 in August 2005). In this version the installation method has changed to be more standard with the “configure; make;

¹A third problem reported at the workshop concerning small differences (up to $\approx 0.5\%$) between the PDFs produced by LHAPDF for MRST and the authors’ code directly is now believed to be due to slight mismatches of grid boundaries at the heavy quark thresholds and will be corrected in future MRST grids.

make install” sequence familiar to many and also a small amount of code has been altered to be more compliant with proprietary FORTRAN 95 compilers. As mentioned in the previous section the web site for public access to LHAPDF from which the source code can be obtained has changed. Since this is the current and most recent version we assume V4.1 in the following referring to earlier versions where necessary.

4 Using LHAPDF

Once the code and PDF data sets have been downloaded from the relevant web site and installed following the instructions given therein, using LHAPDF is simply a matter of linking the compiled FORTRAN library **libLHAPDF.a** to the users program. Table 1 lists the LHAPDF routines available to the user, which are of three types:

- Initialisation (selecting the required PDF set and its member)
- Evolution (producing the momentum density functions (f) for the partons at selected x and Q)
- Information (displaying for example α_s , descriptions, etc.)

Table 1: LHAPDF commands

Command	Description
call InitPDFset (<i>name</i>)	Initialises the PDF set to use.
call InitPDF (<i>member</i>)	Selects the member from the above PDF set.
call evolvePDF (x, Q, f)	Returns the momentum density function, $f(x, Q)$, for protons or pions.
call evolvePDFp ($x, Q, P2, ip2, f$)	Returns the momentum density function for photons ² .
call numberPDF (<i>num</i>)	Returns the number (<i>num</i>) of PDF members in the set.
call GetDesc ()	Prints a description of the PDF set.
alphasPDF (Q)	Function giving the value of α_s at Q GeV.
call GetLam4 (<i>mem, qcdl4</i>)	Returns the value of Λ_4^{QCD} for the specific member.
call GetLam5 (<i>mem, qcdl5</i>)	Returns the value of Λ_5^{QCD} for the specific member.
call GetOrderPDF (<i>order</i>)	Returns the order of the PDF evolution.
call GetOrderAs (<i>order</i>)	Returns the order of the evolution of α_s .
call GetRenFac (<i>muf</i>)	Returns the renormalisation factor.
call GetQmass (<i>nf, mass</i>)	Returns the mass of the parton of flavour <i>nf</i> .
call GetThreshold (<i>nf, Q</i>)	Returns the threshold value for parton of flavour <i>nf</i> .
call GetNf (<i>nfmax</i>)	Returns the number of flavours.

The evolution commands utilise a double precision array $f(-6:6)$ where the arguments range from -6 to +6 for the different (anti)partons as shown in Table 2 below.

Table 2: The flavour enumeration scheme used for $f(n)$ in LHAPDF

parton	\bar{t}	\bar{b}	\bar{c}	\bar{d}	\bar{u}	\bar{d}	g	d	u	s	c	b	t
n	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6

Specifying the location of the PDF sets in the code should be especially mentioned at this point. The argument (*name*) in **InitPDFset** should specify the complete path (or at least to a symbolic link to this

²In **evolvePDFp** $P2$ is the virtuality of the photon in GeV², which should be 0 for an on-shell photon, and $ip2$ is the parameter to evaluate the off-shell anomalous component. See the PDFLIB manual [1] for details.

path). From version 4.1 onwards, however, a new routine **InitPDFsetByName** can be used in which only the name of the PDF set need be specified. This works in conjunction with the script ‘lhpdf-config’ which is generated at the configure stage of the installation which provides the correct path to the PDF sets. The location of this script must therefore be in the users execution path. Tables 3 and 4 list the complete range of PDF set available. The equivalent numbers to use in LHAGLUE, as described in the next section, are also listed in these tables.

Table 3: The Proton PDF sets available in LHAPDF.

Ref	Prefix	Suffix (number of sets)	type	LHAGLUE numbers
[4]	alekhin_	100 (100), 1000 (1000)	p	40100-200, 41000-1999
[5]	a02m_	lo (17), nlo (17), nnlo (17)))	g	40350-67, 40450-67, 40550-67
[6]	botje_	100 (100),1000 (1000)	p	50100-200, 51000-1999
[7]	cteq	61 (41)	p,g	10100-40 , 10150-90
[8]	cteq	6 (41)	p,g	10000-40 , 10050-90
	cteq	6m, 6l, 6ll	p	10000, 10041, 10042
[9]	cteq	5m, 5m1, 5d, 5l	g	19050, 19051, 19060, 19070
[10]	cteq	4m, 4d, 4l	g	19150, 19160, 19170
[11]	fermi2002_	100 (100), 1000 (1000)	p	30100-200, 31000-2000
[12]	GRV98	lo, nlo(2)	g	80060, 80050-1
[13]	H12000	msE (21), disE (21), loE (21)	g	70050-70, 70150-70, 70250-70
[14]	MRST2004	nlo	p,g	20400 , 20450
	MRST2004	nnlo	g	20470
[15]	MRST2003	cnlo	p,g	20300 , 20350
	MRST2003	cnno	g	20370
[16]	MRST2002	nlo (2)	p,g	20200 , 20250
	MRST2002	nnlo	g	20270
	MRST2001	E (31)	p,g	20100-130 , 20150-180
[17]	MRST2001	nlo(4)	p,g	20000-4 , 20500-4
	MRST2001	lo, nnlo	g	20060, 20070
[18]	MRST98	(3)	p	29000-3
	MRST98	lo (5), nlo (5) dis (5), ht	g	29040-5, 29050-5,29060-5,29070-5
[19]	ZEUS2002_	TR (23), FF (23), ZM (23)	p	60000-22, 60100-22, 60200-22
[20]	ZEUS2005_	ZJ (23)	p	60300-22

Notes:

LHAPDF → PrefixSuffix.LHpdf (type p), filename → PrefixSuffix.LHgrid (type g).

Where both p and g are present (p,g) the user has the choice of either.

LHAGLUE numbers in **bold** are the type p (.LHpdf) sets.

5 LHAGLUE

The LHAGLUE interface [21] to LHAPDF is designed along the lines of the existing interface from PYTHIA to PDFLIB.³ For both HERWIG and PYTHIA the existing ‘hooks’ for PDFLIB have been utilised for the LHAGLUE interface. This makes it possible to link it exactly like PDFLIB with no further changes to PYTHIA’s or HERWIG’s source code needing to be implemented.

³DB would like to thank T. Sjöstrand and S. Mrenna for discussions on this topic.

Table 4: The Pion and Photon PDF sets available in LHAPDF.

Prefix	Suffix	LHAGLUE numbers	Prefix	Suffix	LHAGLUE numbers
Pion PDFs			Photon PDFs		
OWPI	(2)	211-12	DOG	0, 1	311, 312
SMRSPI	(3)	231-3	DGG	(4)	321-4
GRVPI	0, 1	251, 252	LACG	(4)	331-4
ABFKWPI	(3)	261-3	GSG	0 (2), 1	341-2, 343
All filenames are PrefixSuffix.LHgrid			GSG96	0, 1	344, 345
The nomenclature used here is essentially the same as in PDFLIB and the relevant publication references can be found in the PDFLIB manual [1].			GRVG	0 (2), 1 (2)	351-2, 353-4
			ACFGPG	(3)	361-3
			WHITG	(6)	381-6
			SASG	(8)	391-8

The interface contains three subroutines (similar to PDFLIB) and can be used seamlessly by Monte Carlo generators interfaced to PDFLIB or in standalone mode. These are described in Table 5. In addition any of the LHAPDF routines, **except** the initialisation routines **InitPDFset** and **InitPDF**, described in Table 1, can also be used, for example to return the value of the strong coupling constant α_s (**alphasPDF(Q)**), or to print the file description (**call GetDesc()**).

There are also several CONTROL switches specified through the 20 element character array LHAPARM and COMMON blocks which determine how the interface operates.

- Location of the LHAPDF library of PDFs (pathname):
From version LHAPDF v4.1 onwards, and the LHAGLUE routines distributed with it, the location of the PDFsets data files is set automatically using the "lhpdf-config" script as described in the previous section, provided that the prescribed installation instructions have been used. For previous versions (4.0 and earlier) the common block **COMMON/LHAPDFC/LHAPATH** is used where **LHAPATH** is a **character*132** variable containing the full path to the PDF sets. The default path is subdir 'PDFsets' of the current directory.
- Statistics on under/over-flow requests for PDFs outside their validity ranges in x and Q^2 .
 - a) **LHAPARM(16) .EQ. 'NOSTAT'** → No statistics (faster)
 - b) **LHAPARM(16) .NE. 'NOSTAT'** → Default: collect statistics
 - c) **call PDFSTA** at the end to print out statistics.
- Option to use the values for α_s as computed by LHAPDF in the Monte Carlo generator as well in order to ensure uniform α_s values throughout a run
 - a) **LHAPARM(17) .EQ. 'LHAPDF'** → Use α_s from LHAPDF
 - b) **LHAPARM(17) .NE. 'LHAPDF'** → Default (same as LHAPDF V1/V3)
- Extrapolation of PDFs outside the LHAPDF validity range given by $x_{min/max}$ and $Q_{min/max}^2$.
 - a) Default → PDFs "frozen" at the boundaries.
 - b) **LHAPARM(18) .EQ. 'EXTRAPOLATE'** → Extrapolate PDFs at own risk
- Printout of initialisation information in PDFSET (by default)
 - a) **LHAPARM(19) .EQ. 'SILENT'** → No printout (silent mode).
 - b) **LHAPARM(19) .EQ. 'LOWKEY'** → Print 5 times (almost silent).
- Double Precision values of $\Lambda_{4/5}^{QCD}$ applicable to the selected PDF are available (as read-only) in the COMMON block: **COMMON/W50512/QCDL4,QCDL5** → as in PDFLIB.

Table 5: LHAGLUE commands

Command	Description
call PDFSET (<i>parm,value</i>)	For initialisation (called once) where PARM and VALUE are LOCAL arrays in the calling program specified as CHARACTER*20 PARM(20) DOUBLE PRECISION VALUE(20)
call STRUCTM (<i>X,Q,UPV,DNV,USEA,DSEA,STR,CHM,BOT,TOP,GLU</i>)	For the proton (and pion) PDFs: where X and Q are the input kinematic variables and the rest are the output PDF of the valence and sea quarks and the gluon.
call STRUCTP (<i>X,Q2,P2,IP2,UPV,DNV,USEA,DSEA,STR,CHM,BOT,TOP,GLU</i>)	For the photon PDFs: as above with the additional input variables P2 and IP2 ² .

The LHAGLUE interface can be invoked in one of 3 ways, Standalone, PYTHIA or HERWIG, depending on the value of *parm(1)* when calling **PDFSET**(*parm,value*).

- **Standalone mode**
 PARM(1)= 'DEFAULT'
 VALUE(1) = "*PDF number*"
- **PYTHIA mode**
 PARM(1) = 'NPTYPE' ← set automatically in PYTHIA
 In this case the user must supply MSTP(51) and MSTP(52) in the PYTHIA common block
COMMON/PYPARS/MSTP(200),PARP(200),MSTI(200).PARI(200)
 MSTP(52) = 2 ← to use an external PDF library
 MSTP(51)= "*PDF number*"
- **HERWIG mode**
 PARM(1) = 'HWLHAPDF' ← set by the user.
 In this case one sets for the beam and target particles separately
 AUTPDF(1) = 'HWLHAPDF'
 AUTPDF(2) = 'HWLHAPDF'
 MODPDF(1) = "*PDF number*"
 MODPDF(2) = "*PDF number*"
 Note that HERWIG specifies the "*PDF number*" for each of the colliding particles separately and care should be taken that the same PDF members are used when appropriate.

The user then simply links their own standalone code, or the HERWIG/PYTHIA main program and the HERWIG/PYTHIA code⁴, with the LHAPDF library **libLHAPDF.a** making sure the 'PDFsets' directory is specified as described above.

The LHAGLUE interface has been tested extensively at TEVATRON and LHC energies for the proton PDFs and with HERA examples for the photon PDFs. Results with new and legacy PDF sets, using LHAPDF, PDFLIB or internal implementations in the Monte Carlo generators, and comparing cross sections produced with PYTHIA and HERWIG, give us confidence in the consistency of the LHAGLUE interface and the underlying LHAPDF library [22].

⁴It is important when starting with a fresh PYTHIA or HERWIG download the user must first rename the 'dummy' sub-routines **STRUCTM**, **STRUCTP** and **PDFSET** in the PYTHIA/ HERWIG source codes exactly as if one were to link to PDFLIB.

6 Summary and Future Development

Both LHAPDF and the interface LHAGLUE have been developed over the period of the Workshop to a point where they can now be used as a serious replacement for PDFLIB. Indeed, except for the PDF authors' own code, they are the only place to obtain the latest PDFs. There is however still considerable development in progress and the latest PDF sets will be incorporated as and when they become available. One major development area is to include the possibility of having more than one PDF set initialised concurrently. This may be necessary in interactions between different beam and target particles types and also including photon and pion PDFs. This will be the aim of the next LHAPDF release.

Acknowledgements

MRW wishes to thank the UK PPARC for support from grant PP/B500590/1. DB wishes to thank the USA National Science Foundation for support under grants NSF ITR-0086044 and NSF PHY-0122557.

References

- [1] H. Plathow-Besch, *Comput. Phys. Commun.* **75**, 396 (1993).
- [2] W. Giele *et al.*, in *The QCD/SM working group: Summary report*. 2002. Also in preprint hep-ph/0204316.
- [3] M. Botje, *QCDNUM version 16.12* (unpublished). Available on <http://www.nikhef.nl/h24/qcdcode/index.html>.
- [4] S. I. Alekhin, *Phys. Rev.* **D63**, 094022 (2001).
- [5] S. I. Alekhin, *Phys. Rev.* **D68**, 014002 (2003).
- [6] M. Botje, *Eur. Phys. J.* **C14**, 285 (2000).
- [7] D. Stump *et al.*, *JHEP* **10**, 046 (2003).
- [8] J. Pumplin *et al.*, *JHEP* **07**, 012 (2002).
- [9] H. L. Lai *et al.*, *Eur. Phys. J.* **C12**, 375 (2000).
- [10] H. L. Lai *et al.*, *Phys. Rev.* **D55**, 1280 (1997).
- [11] W. T. Giele, S. A. Keller and D. A. Kosower, *Parton distribution function uncertainties*. Preprint hep-ph/0104052, 2001.
- [12] M. Glück, E. Reya and A. Vogt, *Eur. Phys. J.* **C5**, 461 (1998).
- [13] C. Adloff *et al.* [H1 Collaboration], *Eur. Phys. J.* **C30**, 1 (2003).
- [14] A. D. Martin, R. G. Roberts, W. J. Stirling and R. S. Thorne, *Eur. Phys. J.* **C39**, 155 (2005).
- [15] A. D. Martin, R. G. Roberts, W. J. Stirling and R. S. Thorne, *Mrst partons and uncertainties*. Preprint hep-ph/0307262, 2003.
- [16] A. D. Martin, R. G. Roberts, W. J. Stirling and R. S. Thorne, *Eur. Phys. J.* **C28**, 455 (2003).
- [17] A. D. Martin, R. G. Roberts, W. J. Stirling and R. S. Thorne, *Eur. Phys. J.* **C23**, 73 (2002).
- [18] A. D. Martin, R. G. Roberts, W. J. Stirling and R. S. Thorne, *Eur. Phys. J.* **C4**, 463 (1998).
- [19] S. Chekanov *et al.* [ZEUS Collaboration], *Phys. Rev.* **D67**, 012007 (2003).
- [20] S. Chekanov *et al.* [ZEUS Collaboration], *An NLO QCD analysis of inclusive cross-section and jet-production data from the ZEUS experiment - DESY-05-050*. (unpublished). Preprint hep-ph/0503274, 2005.
- [21] D. Bourilkov, *Study of parton density function uncertainties with LHAPDF and PYTHIA at LHC* (unpublished). Preprint hep-ph/0305126, 2003.
- [22] DB, CG, MRW. Note in preparation.

THEPEG

Toolkit for High Energy Physics Event Generation

Leif Lönnblad

Department of Theoretical Physics, Lund University, Sweden

Abstract

I present the status of the THEPEG project for creating a common platform for implementing C++ event generators. I also describe briefly the status of the new versions of PYTHIA, HERWIG and ARIADNE which are implemented using this framework.

1 Introduction

Monte Carlo Event Generators (EGs) have developed into essential tools in High Energy Physics. Without them it is questionable if it at all would be possible to embark on large scale experiments such as the LHC. Although the current EGs work satisfactorily, the next generation of experiments will substantially increase the demands both on the physics models implemented in the EGs and on the underlying software technology.

The current EGs are typically written in Fortran and their basic structure was designed almost two decades ago. Meanwhile there has been a change in programming paradigm, towards object oriented methodology in general and C++ in particular. This applies to almost all areas of high-energy physics, but in particular for the LHC program, where all detector simulation and analysis is based on C++. When designing the next generation of EGs it is therefore natural to use C++. Below is a brief description of the THEPEG [1] project for designing a general framework in C++ for implementing EG models, and also the PYTHIA7 and ARIADNE programs which uses THEPEG to implement their respective physics models. Also HERWIG++ is implemented in the THEPEG framework, but this program is described elsewhere in these proceedings [2]

2 Basic structure

THEPEG is a general platform written in C++ for implementing models for event generation. It is made up from the basic model-independent parts of PYTHIA7 [3, 4], the original project of rewriting the Lund family of EGs in C++. When the corresponding rewrite of the HERWIG program [5] started it was decided to use the same basic infrastructure as PYTHIA7 and therefore the THEPEG was factorized out of PYTHIA7 and is now the base of both PYTHIA7 and HERWIG++ [6]. Also the coming C++ version of ARIADNE [7] is using THEPEG.

THEPEG uses CLHEP [8] and adds on a number of general utilities such as smart pointers, extended type information, persistent I/O, dynamic loading and some extra utilities for kinematics, phase space generation etc.

The actual event generation is then performed by calling different *handler* classes for hard partonic sub-processes, parton densities, QCD cascades, hadronization etc. To implement a new model to be used by THEPEG, the procedure is then to write a new C++ class inheriting from a corresponding handler class and implement a number of pre-defined virtual functions. Eg. a class for implementing a new hadronization model would inherit from the abstract `HandronizationHandler` class, and a new parton density parameterization would inherit from the `PDFBase` class.

To generate events with THEPEG one first runs a setup program where an `EventGenerator` object is set up to use different models for different steps of the generation procedure. All objects to be chosen from are stored in a *repository*, within which it is also possible to modify switches and parameters of the implemented models in a standardized fashion, using so called *interface* objects. Typically the user would choose from a number of pre-defined `EventGenerator` objects and only make minor changes for the specific simulation to be made. When an `EventGenerator` is properly set up it is saved persistently to a file which can then be read into a special run program to perform the generation, in which case special `AnalysisHandler` objects may be specified to analyze the resulting events. Alternatively it can be read into eg. a detector simulation program or a user supplied analysis program, where it can be used to generate events.

3 Status

3.1 THEPEG

THEPEG version 1.0 α is available [1] and is working. As explained above, it contains the basic infrastructure for implementing and running event generation models. It also contains some simple physics models, such as some $2 \rightarrow 2$ matrix elements, a few parton density parameterizations and a near-complete set of particle decays. However, these are mainly in place for testing purposes, and to generate realistic events, the PYTHIA7 and/or HERWIG++ programs are needed.

Currently the program only works under Linux using the gcc compiler. This is mainly due to the use of dynamic linking of shared object files, which is inherently platform-dependent. Recently, the build procedure has been redesigned using the `libtool` facility [9], which should allow for easy porting to other platforms in the future.

Although THEPEG includes a general structure for implementing basic fixed-order matrix element generation to produce the initial hard subprocesses in the event generation, a general procedure for reading such parton level events from external programs using the Les Houches accord [10] has been developed and will be included in the next release¹.

The documentation of THEPEG is currently quite poor. Recently the actual code documentation was converted to Doxygen format [11], which will hopefully facilitate the documentation process. The lack of documentation means that there is currently a fairly high threshold for a beginner to start using and/or developing physics modules for THEPEG. The situation is further complicated since the user interface is currently quite primitive. THEPEG has a well worked through low-level interface to be able to set parameter and switches, etc. in classes introduced to the structure from the outside. However, the current external user interface is a simple command-line facility which is not very user-friendly. A Java interface is being worked on, but is not expected to be released until next year.

3.2 PYTHIA 7 (and PYTHIA8)

PYTHIA7 version 1.0 α is available [4] and is working. It contains a reimplementations of the parton shower and string fragmentation models currently available in the 6.1 version of PYTHIA [12]. In an unfortunate turn of events, the principal PYTHIA author, Torbjörn Sjöstrand, has decided to leave the THEPEG collaboration and is currently developing a new C++ version of PYTHIA (called PYTHIA8 [13]) on his own. This means that the development of PYTHIA7 is stopped, but hopefully it will be possible to interface the different modules in PYTHIA8 so that they can be used within the general framework of THEPEG.

¹A snapshot of the current development version is available from [1]

3.3 ARIADNE

The reimplementaion of the ARIADNE [7, 14] program using the framework of THEPEG has just started and is, hence, not publically available yet. Although this is mainly a pure rewrite of the fortran version of ARIADNE, it will contain some improvements, such as the CKKW matching [15, 16] also planned for HERWIG++. In addition, an improved version of the LDCMC [17] is planned.

4 Conclusions

THEPEG was intended to be *the* standard platform for event generation for the LHC era of collider physics. Unfortunately, this does not seem to become a reality. Besides the recent split between PYTHIA and THEPEG, there will also be other separate programs such as SHERPA [18, 19]. This is, of course, not an optimal situation, especially not for the LHC experiments, which surely would have preferred a uniform interface to different event generator models.

References

- [1] L. Lönnblad *et al.*, *THEPEG program*. <http://www.thep.lu.se/ThePEG>.
- [2] S. Gieseke, *HERWIG++*. These proceedings.
- [3] M. Bertini, L. Lönnblad, and T. Sjöstrand, *Comput. Phys. Commun.* **134**, 365 (2001). [hep-ph/0006152](http://arxiv.org/abs/hep-ph/0006152).
- [4] L. Lönnblad *et al.*, *PYTHIA7 program*. <http://www.thep.lu.se/Pythia7>.
- [5] G. Corcella *et al.*, *JHEP* **01**, 010 (2001). [hep-ph/0011363](http://arxiv.org/abs/hep-ph/0011363).
- [6] S. Gieseke, A. Ribon, M. H. Seymour, P. Stephens, and B. Webber, *JHEP* **02**, 005 (2004). [hep-ph/0311208](http://arxiv.org/abs/hep-ph/0311208).
- [7] L. Lönnblad, *Comput. Phys. Commun.* **71**, 15 (1992).
- [8] L. Lönnblad, *Comput. Phys. Commun.* **84**, 307 (1994).
- [9] G. Matzigkeit *et al.*, *libtool program*. <http://www.gnu.org/software/libtool>.
- [10] E. Boos *et al.* (2001). [hep-ph/0109068](http://arxiv.org/abs/hep-ph/0109068).
- [11] D. van Heesch, *The doxygen documentation system*. <http://www.doxygen.org>.
- [12] T. Sjöstrand, and others, *Comput. Phys. Commun.* **135**, 238 (2001). [arXiv:hep-ph/0010017](http://arxiv.org/abs/hep-ph/0010017).
- [13] T. Sjöstrand, *PYTHIA*. These proceedings.
- [14] L. Lönnblad, *ARIADNE at HERA and at the LHC*. These proceedings.
- [15] S. Catani, F. Krauss, R. Kuhn, and B. R. Webber, *JHEP* **11**, 063 (2001). [hep-ph/0109231](http://arxiv.org/abs/hep-ph/0109231).
- [16] L. Lönnblad, *JHEP* **05**, 046 (2002). [hep-ph/0112284](http://arxiv.org/abs/hep-ph/0112284).
- [17] H. Kharraziha and L. Lönnblad, *JHEP* **03**, 006 (1998). [hep-ph/9709424](http://arxiv.org/abs/hep-ph/9709424).
- [18] T. Gleisberg *et al.*, *JHEP* **02**, 056 (2004). [hep-ph/0311263](http://arxiv.org/abs/hep-ph/0311263).
- [19] T. Gleisberg *et al.*, *The event generator SHERPA*. These proceedings.

PYTHIA

T. Sjöstrand

CERN, Geneva, Switzerland, and
Department of Theoretical Physics, Lund University, Sweden

The PYTHIA program is a standard tool for the generation of high-energy collisions, containing a realistic description of the full story, from a hard interaction involving a few partons to an observable hadronic final state of hundreds of particles. The current PYTHIA 6.3 version is described in detail in the manual [1], with the most recent update notes to be found on the PYTHIA webpage <http://www.thep.lu.se/~torbjorn/Pythia.html>, together with the code itself, sample main programs and some further material. The latest published version is [2] and a recent brief review is found in [3]. The 6.3 version includes new transverse-momentum-ordered showers and a new multiple-interactions and beam-remnant scenario [4], described elsewhere in these proceedings.

From the onset, all PYTHIA code has been written in Fortran 77. For the LHC era, the experimental community has made the decision to move heavy computing completely to C++. Hence the main future development line is PYTHIA 8, which is a re-implementation in C++. Many obsolete options will be removed and various aspects modernized in the process.

With the rise of automatic matrix-element code generation and phase-space sampling, input of process-level events via the Les Houches Accord (LHA) [5] reduces the need to have extensive process libraries inside PYTHIA itself. Thus emphasis is on providing a good description of subsequent steps of the story, involving elements such as initial- and final-state parton showers, multiple parton-parton interactions, string fragmentation, and decays. All the latter components now exist as C++ code, even if in a preliminary form, with finer details to be added, and still to be better integrated and tuned. At the current stage, however, there is not even the beginning of a PYTHIA 8 process library; instead a temporary interface is provided to PYTHIA 6, so that all hard processes available there can be generated and sent on to PYTHIA 8, transparent to the user.

PYTHIA 8 is intended to be a standalone program, i.e. does not require any external libraries. However, in addition to the LHA interface, hooks also exist for external parton distribution functions, particle decays and random numbers, and more may be added.

This project was started in September 2004, and so is still at an early stage. A first public version, PYTHIA 8.040, can be found on the PYTHIA webpage (look under the “Future” link). This should be viewed as a development snapshot, to allow early feedback from the LHC experimental community, and cannot be used for any serious physics studies. It is intended/hoped that a first realistic version, PYTHIA 8.100, could be ready by early 2007, but even this version will be clearly limited in its capabilities, and strongly focused on LHC applications. It is therefore to be expected that PYTHIA 6 and PYTHIA 8 will co-exist for several years.

References

- [1] T. Sjöstrand, L. Lönnblad, S. Mrenna, and P. Skands (2003). [hep-ph/0308153](#).
- [2] T. Sjöstrand *et al.*, *Comput. Phys. Commun.* **135**, 238 (2001). [hep-ph/0010017](#).
- [3] M. A. Dobbs *et al.* (2004). [hep-ph/0403045](#).
- [4] T. Sjöstrand and P. Z. Skands, *JHEP* **03**, 053 (2004). [hep-ph/0402078](#);
T. Sjöstrand and P. Z. Skands, *Eur. Phys. J.* **C39**, 129 (2005). [hep-ph/0408302](#).
- [5] E. Boos *et al.* (2001). [hep-ph/0109068](#).

HERWIG

Michael H. Seymour

School of Physics & Astronomy, University of Manchester, and Theoretical Physics Group, CERN

Abstract

I review the status of the current fortran version of HERWIG. Progress towards its replacement, Herwig++, is reviewed elsewhere in these proceedings.

1 Introduction

HERWIG [1] is a Monte Carlo event generator for simulation of hadronic final states in lepton–lepton, lepton–hadron and hadron–hadron collisions. It incorporates important colour coherence effects in the final state [2] and initial state [3] parton showers, as well as in heavy quark processes [4] and the hard process generation [5]. It uses the cluster [6] hadronization model and a cluster-based simulation of the underlying event [7]. While earlier versions [8] concentrated on QCD and a few other SM processes, recent versions contain a vast library of MSSM [9] and other BSM processes. A review of current Monte Carlo event generators including HERWIG can be found in [10].

We are currently in a period of intense activity, finalizing the HERWIG program and writing a completely new event generator, HERWIG++. In this very short contribution, I can do little more than mention the areas of progress and provide references to sources of more details.

2 HERWIG version 6.5

HERWIG version 6.5 was released [11] in October 2002. Its main new features were an interface to the Les Houches Accord event format [12], the hooks needed by the MC@NLO package [13] and various bug fixes and minor improvements. It was advertised as the final fortran version of HERWIG before work switched to HERWIG++.

Despite this, the period since then has seen intense development with several new subversion releases and new features, most notably version 6.505, which featured an improved interface to the Jimmy generator for multiparton interactions, which I will discuss in more detail shortly. The most recent version is 6.507, which can be obtained from the HERWIG web site [14].

Development of fortran HERWIG is now slowing, and the only new feature still foreseen is the implementation of matrix element corrections to the production of Higgs bosons, both SM and MSSM, preliminary versions of which have been discussed in [15]. Beyond this, the HERWIG collaboration has made a commitment to all running (and ceased) experiments to support their use of HERWIG throughout their lifetimes. Due to lack of manpower, making the same promise to the LHC experiments would divert too much effort away from support of HERWIG++, and we will only support their use of HERWIG until we believe that HERWIG++ is a stable alternative for production running.

3 Jimmy

Early versions of the Jimmy model [16] generated jet events in photoproduction using a multiparton interaction picture. The recent update [17] enables it to work efficiently as a generator of underlying events in high E_T jet events and other hard processes in hadron–hadron collisions for the first time. For a given pdf set, the main adjustable parameters are PTJIM, the minimum transverse momentum of partonic scattering, and JMRAD(73), related to the effective proton radius. Varying these one is able to get a good description of the CDF data [18] and other data held in the JetWeb database [19] that are sensitive to underlying event effects in hard process events. However, a poor description of minimum bias data in which there is no hard scale is still obtained. This is probably due to the fact that PTJIM is a hard cutoff

and there is no soft component below it; preliminary attempts to rectify this are encouraging [20]. It is interesting to note that with tunings that give equally good descriptions of current data, Jimmy predicts twice as much underlying event activity as PYTHIA at the LHC.

References

- [1] G. Corcella *et al.*, JHEP **0101** (2001) 010 [arXiv:hep-ph/0011363].
- [2] G. Marchesini and B.R. Webber, Nucl. Phys. B **238** (1984) 1.
- [3] G. Marchesini and B.R. Webber, Nucl. Phys. B **310** (1988) 461.
- [4] G. Marchesini and B.R. Webber, Nucl. Phys. B **330** (1990) 261.
- [5] R.K. Ellis, G. Marchesini and B.R. Webber, Nucl. Phys. B **286** (1987) 643 [Erratum-ibid. B **294** (1987) 1180].
- [6] B.R. Webber, Nucl. Phys. B **238** (1984) 492.
- [7] G. Marchesini and B.R. Webber, Phys. Rev. D **38** (1988) 3419.
- [8] G. Marchesini, B.R. Webber, G. Abbiendi, I.G. Knowles, M.H. Seymour and L. Stanco, Comput. Phys. Commun. **67** (1992) 465.
- [9] S. Moretti, K. Odagiri, P. Richardson, M.H. Seymour and B.R. Webber, JHEP **0204** (2002) 028 [arXiv:hep-ph/0204123].
- [10] M.A. Dobbs *et al.*, “Les Houches guidebook to Monte Carlo generators for hadron collider physics”, *Workshop on Physics at TeV Colliders, Les Houches, France, 26 May–6 June 2003*, arXiv:hep-ph/0403045.
- [11] G. Corcella *et al.*, “HERWIG 6.5 release note”, arXiv:hep-ph/0210213.
- [12] E. Boos *et al.*, “Generic user process interface for event generators”, *Workshop on Physics at TeV Colliders, Les Houches, France, 21 May–1 June 2001*, arXiv:hep-ph/0109068.
- [13] S. Frixione and B.R. Webber, JHEP **0206** (2002) 029 [arXiv:hep-ph/0204244];
S. Frixione and B.R. Webber, “The MC@NLO event generator”, arXiv:hep-ph/0207182;
S. Frixione, P. Nason and B.R. Webber, JHEP **0308** (2003) 007 [arXiv:hep-ph/0305252].
- [14] <http://hepwww.rl.ac.uk/theory/seymour/herwig/> also with a link to Jimmy’s web page.
- [15] G. Corcella and S. Moretti, Phys. Lett. B **590** (2004) 249 [arXiv:hep-ph/0402146];
G. Corcella and S. Moretti, “Matrix-element corrections to $gg/q\bar{q} \rightarrow$ Higgs in HERWIG”, *Workshop on Physics at TeV Colliders, Les Houches, France, 26 May–6 June 2003*, arXiv:hep-ph/0402149.
- [16] J.M. Butterworth, J.R. Forshaw and M.H. Seymour, Z. Phys. C **72** (1996) 637 [arXiv:hep-ph/9601371].
- [17] C.M. Buttar *et al.*, these proceedings.
J.M. Butterworth and M.H. Seymour, in preparation.
- [18] T. Affolder *et al.* [CDF Collaboration], Phys. Rev. D **65** (2002) 092002.
- [19] J.M. Butterworth and S. Butterworth, Comput. Phys. Commun. **153** (2003) 164 [arXiv:hep-ph/0210404].
- [20] I. Borozan and M.H. Seymour, JHEP **0209** (2002) 015 [arXiv:hep-ph/0207283].

Herwig++

Stefan Gieseke

Institut für Theoretische Physik, Universität Karlsruhe, 76128 Karlsruhe, Germany

Abstract

I briefly review the status of the Herwig++ event generator. Current achievements are highlighted and a brief summary of future plans is given.

1 Introduction

Herwig++ [1] is a new Monte Carlo event generator for simulating collider physics, written in the object oriented programming language C++. The idea is to rewrite the well-established multi-purpose event generator HERWIG [2] and to improve it where necessary [3]. The Lund event generators PYTHIA [4] and ARIADNE [5] are also being rewritten at the moment. Herwig++ and ARIADNE will both be based on a common event generation framework, called ThePEG [6] which will make it possible to exchange single modules of the event generation and allows us to have a common, or at least a very similar user interface. PYTHIA8, the rewrite of PYTHIA (6.3) will be written independently of this project but may become integrated into the structure later [7]. A further object oriented event generator, SHERPA [8], is established as an independent project.

2 Event Generation

In its present version (1.0) Herwig++ is capable of simulating e^+e^- annihilation events. The physics simulation consists of several steps, going from small (perturbative) to large (non-perturbative) distance scales. First, the effective CM energy of the annihilating e^+e^- pair is selected according to some model structure function of the electron, thereby radiating photons that carry some fraction of the original energy. Next, we set up the $q\bar{q}$ final state and a hard matrix element correction is applied [9]. In the next step, parton showers are radiated from the coloured final state particles. These effectively resum large soft and collinear logarithms. The parton shower is modelled in terms of new evolution variables with respect to the FORTRAN program [10]. This, and the use of splitting functions for massive particles allow us to simulate the suppression of soft and collinear radiation from heavy particles dynamically (dead cone effect) which has previously only been modelled in a crude way. Parton showers from initial state particles in a hard scattering and from decays of heavy particles (particularly t -quarks) have been formulated for various situations in [10]. The next stage of the simulation is the hadronization of the outgoing coloured particles. First, remaining gluons are split into non-perturbative $q\bar{q}$ -pairs. Colour connected particles are paired into colourless clusters. The invariant mass spectrum of these clusters contains a long high-mass tail that still contains a large scale. These heavy clusters are further split into pairs of lighter clusters. Once all clusters are below a certain mass threshold they decay into pairs of hadrons. The hadron species are selected only according to a handful of parameters. It is this stage where it has been observed in previous versions of the FORTRAN program that the meson/baryon number ratio in e^+e^- annihilation events was difficult to obtain when a large number of highly excited mesons is available in the program [11]. In the current version the hadron selection is reorganised and we obtain more stable results. Finally, the produced hadrons decay into stable hadrons according to some models. In version 1.0 the hadronic decays were modelled similarly to the decays in the FORTRAN version.

We have tested the simulation of e^+e^- annihilation events in very great detail [1]. We considered event shape variables, jet rates, hadron yields and many more observables. One observable of special interest has been the b -quark fragmentation function that we found to be well-described on the basis of the parton shower only. This is a result of the new shower algorithm for heavy quarks. The overall result was that we are capable of simulating e^+e^- events at least as well good as with the FORTRAN version.

3 Current and Future Developments

Many new features are currently being implemented for the event simulation at hadron colliders. The list of hard matrix elements will be slightly extended in the next version in order to cover some basic processes. In principle we can also rely on parton level event generators and read in event files that follow the Les Houches Accord [12]. The parton shower will include initial state radiation and gluon radiation in the perturbative decay of heavy particles. Some related aspects of estimating uncertainties from initial state parton showers were addressed in [13]. A large effort went into remodelling and updating the secondary hadronic decays. A future version should also be able to simulate hard jets in deep inelastic scattering. Exhaustive tests of our generator output against current data from the experiments at HERA and the Tevatron will be made in order to validate and understand our program. In the long-term we plan to include a larger number of simple processes, mainly $2 \rightarrow 2$ and some $2 \rightarrow 3$, both Standard Model processes and some BSM processes as well. The modelling of the underlying event will at first only be on the basis of the simple so-called UA5 model that is also available in the FORTRAN version. A refinement towards a more sophisticated multiple interaction model [14, 15] is planned.

References

- [1] S. Gieseke, A. Ribon, M. H. Seymour, P. Stephens and B. Webber, JHEP **0402**, 005 (2004) [arXiv:hep-ph/0311208].
- [2] G. Corcella *et al.*, arXiv:hep-ph/0210213; G. Corcella *et al.*, JHEP **0101** (2001) 010 [arXiv:hep-ph/0011363].
- [3] S. Gieseke, in Proc. *Hadron Collider Physics 2002*, eds. M. Erdmann, Th. Müller [arXiv:hep-ph/0210294].
- [4] T. Sjostrand, L. Lonnblad, S. Mrenna and P. Skands, arXiv:hep-ph/0308153; T. Sjostrand, P. Eden, C. Friberg, L. Lonnblad, G. Miu, S. Mrenna and E. Norrbin, Comput. Phys. Commun. **135** (2001) 238 [arXiv:hep-ph/0010017].
- [5] L. Lonnblad, Comput. Phys. Commun. **71** (1992) 15.
- [6] M. Bertini, L. Lonnblad and T. Sjostrand, Comput. Phys. Commun. **134** (2001) 365 [arXiv:hep-ph/0006152].
- [7] T. Sjöstrand, this workshop.
- [8] T. Gleisberg, S. Hoche, F. Krauss, A. Schaliche, S. Schumann and J. C. Winter, JHEP **0402** (2004) 056 [arXiv:hep-ph/0311263].
- [9] M. H. Seymour, Comput. Phys. Commun. **90** (1995) 95 [arXiv:hep-ph/9410414].
- [10] S. Gieseke, P. Stephens and B. Webber, JHEP **0312**, 045 (2003) [arXiv:hep-ph/0310083].
- [11] A. Kupco, in Proc. *Monte Carlo generators for HERA physics*, (Hamburg 1998–1999), eds. A.T. Doyle, G. Grindhammer, G. Ingelman, H. Jung. Hamburg, DESY-PROC-1999-02, 292 [arXiv:hep-ph/9906412].
- [12] E. Boos *et al.*, arXiv:hep-ph/0109068.
- [13] S. Gieseke, JHEP **0501**, 058 (2005) [arXiv:hep-ph/0412342].
- [14] J. M. Butterworth, J. R. Forshaw and M. H. Seymour, Z. Phys. C **72** (1996) 637 [arXiv:hep-ph/9601371].
- [15] I. Borozan and M. H. Seymour, JHEP **0209**, 015 (2002) [arXiv:hep-ph/0207283].

The Event Generator SHERPA

T. Gleisberg, S. Höche, F. Krauss, A. Schälicke, S. Schumann, J. Winter
Institut für theoretische Physik, TU Dresden, D-01062 Dresden, Germany
E-mail: steffen@theory.phy.tu-dresden.de

Abstract

In this contribution the multi-purpose event generation framework SHERPA is presented and the development status of its physics modules is reviewed. In its present version, SHERPA is capable of simulating lepton-lepton, lepton-photon, photon-photon and fully hadronic collisions, such as proton-proton reactions.

SHERPA [1] is an independent approach for a framework for event generation at high energy collider experiments. The program is entirely written in the object-oriented programming language C++. This is reflected in particular in the structure of the program. In SHERPA, the various tasks related to the generation of events are encapsulated in a number of specific modules. These physics modules are initialized and steered by the SHERPA framework. This structure facilitates a high modularity of the actual event generator and allows for the easy replacement/modification of entire physics models, e.g. the parton shower or the fragmentation model. The current version SHERPA-1.0.6 incorporates the following physics modules:

- ATOOLS-2.0.6: This is the toolbox for all other modules. ATOOLS contain classes with mathematical tools like vectors and matrices, organization tools such as read-in or write-out devices, and physics tools like particle data or classes for the event record.
- BEAM-1.0.6: This module manages the treatment of the initial beam spectra for different colliders. At the moment two options are implemented for the beams: they can either be monochromatic, and therefore require no extra treatment, or, for the case of an electron collider, laser backscattering off the electrons is supported leading to photonic initial states.
- PDF-1.0.6: In this module the handling of initial state radiation (ISR) is located. It provides interfaces to various proton and photon parton density functions, and to the LHAPDFv3 interface. In addition, an analytical electron structure function is supplied.
- MODEL-1.0.6: This module comprises the basic physics parameters (like masses, mixing angles, etc.) of the simulation run. Thus it specifies the corresponding physics model. Currently three different physics models are supported: the Standard Model (SM), its Minimal Supersymmetric extension (MSSM) and the ADD model of large extra dimensions. For the input of MSSM spectra a run-time interface to the program *Isasusy 7.67* [2] is provided. The next release of SHERPA will in addition support the SLHA format of spectrum files [3].
- EXTRA_XS-1.0.6: In this module a collection of analytic expressions for simple $2 \rightarrow 2$ processes within the SM and the corresponding classes embedding them into the SHERPA framework are provided. This includes methods used for the definition of the parton shower evolution, such as color connections and the hard scale of the process. The classes for phase space integration, which are common with AMEGIC, are located in a special module called PHASIC.
- AMEGIC++-2.0.6: AMEGIC [4] is SHERPA's own matrix element generator. It works as a generator-generator: during the initialization run the matrix elements for a set of given processes within the SM, the MSSM or the ADD model, as well as their specific phase space mappings are created by AMEGIC and stored in library files. In the initialization of the production run, these libraries are linked to the program. They are used to calculate cross sections and to generate single events based on them.

- PHASIC++-1.0.6: Here all classes dealing with the Monte Carlo phase space integration are located. As default the adaptive multi-channel method of [5] together with a Vegas optimization [6] for the single channels is used for the evaluation of the initial state and final state integrals.
- APACIC++-2.0.6: APACIC [7] contains classes for the simulation of both the initial and the final state parton shower. The sequence of parton emissions in the shower evolution is organized in terms of the parton's virtual mass as ordering parameter. Coherence effects are accounted for by explicit ordering of the opening angles in subsequent branchings. This treatment is similar to the Pythia [8] parton shower approach. All features for a consistent merging with matrix elements [9] are included.
- AMISIC++-1.0.6: AMISIC contains classes for the simulation of multiple parton interactions according to [10]. SHERPA extends this treatment of multiple interactions by allowing for the simultaneous evolution of an independent parton shower in each of the subsequent (semi-)hard collisions. This shower evolution is done by the APACIC module.
- SHERPA-1.0.6: Finally, SHERPA is the steering module that initializes, controls and evaluates the different phases in the entire process of event generation. Furthermore, all necessary routines for combining the parton showers and matrix elements, which are independent of the specific parton shower are found in this module. In addition, this subpackage provides an interface to the Lund String Fragmentation of Pythia 6.214 including its hadron decay routines.

SHERPA is publicly available from <http://www.sherpa-mc.de>. It has successfully been tested for various processes of great relevance at future colliders [11]. Present activities of developing SHERPA cover the modeling of the underlying event and an alternative fragmentation model [12].

References

- [1] Gleisberg, T. and others, JHEP **02**, 056 (2004).
- [2] Paige, F. E. and Protopescu, S. D. and Baer, H. and Tata, X., hep-ph/0312045.
- [3] Skands, P. and others, JHEP **07**, 036 (2004).
- [4] Krauss, F. and Kuhn, R. and Soff, G., JHEP **02**, 044 (2002).
- [5] Kleiss, R. and Pittau, R., Comput. Phys. Commun. **83**, 141 (1994);
Berends, F. A. and Pittau, R. and Kleiss, R., Nucl. Phys. **B424**, 308 (1994).
- [6] Lepage, G. P. CLNS-80/447.
- [7] Kuhn, R. and Krauss, F. and Ivanyi, B. and Soff, G., Comput. Phys. Commun. **134**, 223 (2001);
Krauss, F. and Schällicke, A. and Soff, G., hep-ph/0503087.
- [8] Sjöstrand, T., Comput. Phys. Commun. **82**, 74 (1994);
Sjöstrand, T. and Lönnblad, L. and Mrenna, S. and Skands, P., hep-ph/0308153.
- [9] Catani, S. and Krauss, F. and Kuhn, R. and Webber, B. R., JHEP **11**, 063 (2001);
Krauss, F., JHEP **08**, 015 (2002).
- [10] Sjöstrand, T. and van Zijl, M., Phys. Rev. **D36**, 2019 (1987).
- [11] Gleisberg, T. and others, JHEP **09**, 001 (2003);
Gleisberg, T. and others, Eur. Phys. J. **C34**, 173 (2004);
Krauss, F. and Schällicke, A. and Schumann, S. and Soff, G., Phys. Rev. **D70**, 114009 (2004);
Krauss, F. and Schällicke, A. and Schumann, S. and Soff, G., hep-ph/0503280;
Gleisberg, T. and Krauss, F. and Schällicke, A. and Schumann, S. and Winter, J., hep-ph/0504032.
- [12] Winter, J. and Krauss, F. and Soff, G., Eur. Phys. J. **C36**, 381 (2004).

ARIADNE at HERA and at the LHC

Leif Lönnblad

Department of Theoretical Physics, Lund University, Sweden

Abstract

I describe briefly the status of the ARIADNE program implementing the Dipole Cascade Model and comment both on its performance at HERA, and the uncertainties relating to the extrapolation to LHC energies.

1 Introduction

ARIADNE [1] is a Fortran subroutine library to be used with the PYTHIA event generator [2]. By simply adding a few lines to a PYTHIA steering routine, the PYTHIA parton shower is replaced by the dipole cascade in ARIADNE. For lepton–hadron DIS it can also be used together with the LEPTO [3] generator in a similar fashion. However, even if it thus simple to use ARIADNE also for the LHC, there are a few caveats of which the user should be aware. In this brief presentation of the program, I will first go through the main points of the final-state dipole shower relevant for e^+e^- -annihilation, then I will present the extension of the model to lepton–hadron DIS, and finally describe how the model works for hadron–hadron collisions.

2 The Basic Dipole Model

In the Dipole Cascade Model (DCM) [4, 5], the bremsstrahlung of gluons is described in terms of radiation from colour dipoles between gluons and quarks. Thus, in an $e^+e^- \rightarrow q\bar{q}$ event, a gluon, g_1 may be emitted from the colour-dipole between the q and \bar{q} . In this emission the initial dipole is replaced by two new ones, one between q and g_1 and one between g_1 and \bar{q} . These may then continue radiating independently in a cascade where each step is a $2 \rightarrow 3$ partonic splitting or, equivalently, a splitting of a dipole into two. The splittings are ordered in a transverse momentum variable, p_\perp , defined in a Lorentz-invariant fashion, which also defines the scale in α_S .

There are several advantages of this model. One is that the coherence effects approximated by angular ordering [6] in eg. the HERWIG [7] parton cascade, are automatically included. Another is that the first order $e^+e^- \rightarrow qg\bar{q}$ matrix element correction is in some sense built-in. A major *disadvantage* is that the $g \rightarrow q\bar{q}$ splitting does not enter naturally in this formalism. Final-state $g \rightarrow q\bar{q}$ splittings are, however, easy to add [8] and for final-state cascades in e^+e^- -annihilation the description is complete. Ariadne is generally considered to be the program which best reproduces event shapes and other hadronic final-state observables at LEP (see eg. [9]).

3 ARIADNE at HERA

While for e^+e^- -annihilation, the DCM is formally equivalent to conventional angular ordered parton showers to modified leading logarithmic accuracy, the situation for collisions with incoming hadrons is quite different. In a conventional shower the struck quark in eg. lepton–hadron DIS is evolved backwards with an initial-state cascade according to DGLAP [10–13] evolution. In contrast, the DCM model for DIS [14] describes all gluon emissions in terms of final-state radiation from colour-dipoles, in a similar way as in e^+e^- -annihilation, with the initial dipole now being between the struck quark and the hadron remnant. Contrary to e^+e^- -annihilation, the remnant must now be treated as an extended object and, since radiation of small wavelengths from an extended antenna is suppressed, the emission of high- p_\perp gluons in the DCM is suppressed in the remnant direction.

Despite this suppression, which is modeled semi-classically, the net result is that gluon emissions are allowed in a much larger phase space region than in a conventional parton shower, especially for limited Q^2 values. Although the emissions are ordered in p_\perp , they are not ordered in rapidity (or x). Hence, if tracing the emissions in rapidity, they will be unordered in p_\perp , and there are therefore qualitative similarities between the DCM and BFKL evolution [15–17]. This is in contrast to conventional showers which are purely DGLAP-based and where the emissions are ordered both in scale and in x . One of the striking phenomenological consequences of this is that ARIADNE is one of the few programs which are able to describe the high rate of forward (in the proton direction) jets measured in small- x DIS at HERA [18–20], an observable which conventional parton showers completely fail to reproduce. In fact, ARIADNE is in general considered to be the program which best describe hadronic final-state observables at HERA [21].

This does not mean that the DCM is perfect in any way. Most notably, the initial-state $g \rightarrow q\bar{q}$ and $q \rightarrow g^*q$ (where the q is emitted into the final-state) splittings are not easily included. While the former process has been included as an explicit initial-state splitting step [22], the latter is currently absent in the the ARIADNE program. In addition, the treatment of the initial-state $g \rightarrow q\bar{q}$ splitting has been found to be somewhat incomplete, as it by construction imposes ordering in both p_\perp and rapidity, thus excluding certain regions of the allowed phase space. At HERA, the incomplete treatment of the $g \rightarrow q\bar{q}$ and $q \rightarrow g^*q$ splittings can be shown to be a small effect. However, this is not always the case at the LHC.

4 ARIADNE at LHC

Given the great success of ARIADNE at LEP and HERA, it is natural to assume that it also would do a good job at the Tevatron and the LHC. In principle, the extension of the DCM to hadron–hadron collisions is trivial, and indeed it is simple to run ARIADNE together with PYTHIA for hadron–hadron collisions. Whichever hard sub-process, PYTHIA generates, the relevant dipoles between hard partons and hadron remnants are constructed and are allowed to radiate. In addition, the initial-state $g \rightarrow q\bar{q}$ splittings are included from both sides. However, for many processes there are modifications needed.

The most obvious processes are Drell-Yan and vector boson production, where a quark from one hadron annihilates with an anti-quark from the other. The gluon radiation is then initiated by the dipole between the two remnants, and we have a suppression in both directions. However, it is unphysical to give the remnants a large transverse momentum from the recoil of the gluon emission. In DIS, this is resolved by introducing so-called recoil gluons [14], but here it is clear that the recoil should be taken by the vector boson or the Drell-Yan lepton pair. Such a procedure was introduced in [23], and together with a correction where the first emission is matched to the $qg \rightarrow qZ$ and $q\bar{q} \rightarrow gZ$ matrix elements, it describes well eg. the $Z^0 p_\perp$ spectrum measured at the Tevatron [24,25]. There are still some differences wrt. conventional parton showers. Eg. the rapidity correlation between the vector boson and the hardest jet is more flat in ARIADNE due to the increased phase space for emissions [26]. Although W and Z^0 production at the Tevatron is not a small- x process, the effect is related to higher rate of forward jets in ARIADNE for DIS. Such correlations have not yet been measured at the Tevatron, but another related effect is the somewhat harder p_\perp -spectrum of the Z^0 for low p_\perp in ARIADNE, which is compatible with Tevatron measurements [26]. For a conventional cascade to be able to describe the low- p_\perp spectrum, a quite substantial “non-perturbative” intrinsic transverse momentum must be added to the incoming quarks [27,28].

Going from the Tevatron to the LHC, there is a substantial increase in phase space for QCD radiation, and it can be argued that W and Z^0 production at the LHC is a small- x process with $x \propto m_Z/\sqrt{S} < 0.01$. Indeed ARIADNE predicts a harder p_\perp -spectrum for the W at the LHC as compared to conventional showers [29].

Also Higgs production can be argued to be almost a small- x process at the LHC, if the Higgs is found with a mass around the “most likely” value of ≈ 120 GeV. However, Higgs production is a

gluon-initiated process, and the absence of the $q \rightarrow g^*q$ splitting is a serious deficiency giving a much softer p_\perp -spectrum for the Higgs in ARIADNE as compared to conventional showers [30]. Hence the predictions from ARIADNE for this and similar processes can currently not be trusted. Furthermore, the increased phase-space at the LHC means that predictions also for quark-initiated processes may become affected by the deficiencies in the treatment of initial-state $g \rightarrow q\bar{q}$ mentioned above.

5 Conclusion

The success of the DCM as implemented in ARIADNE in describing hadronic final-state observables as measured at LEP and HERA makes it tempting to use it also to make predictions for the LHC. The temptation is even more difficult to resist as it is so simple to run ARIADNE together with PYTHIA for any LHC process. Currently, this must be done with great care. As explained above, it is possible to obtain reasonable predictions for vector boson production. Also standard jet-production should be fairly safe. However, for Higgs production, one of the most interesting processes at LHC, ARIADNE in its current state turns out to be quite useless.

ARIADNE is currently being rewritten in C++ within the framework of THEPEG [31, 32]. The planned features includes a remodeling of initial-state $g \rightarrow q\bar{q}$ splittings as well as the introduction of the $q \rightarrow g^*q$ process. In addition the matching to fixed-order tree-level matrix elements à la CKKW [26, 33–35] will be implemented for the most common sub-processes. When this version is released, hopefully during 2006, it should therefore be safe to use ARIADNE to produce LHC predictions.

References

- [1] L. Lönnblad, *Comput. Phys. Commun.* **71**, 15 (1992).
- [2] T. Sjöstrand, and others, *Comput. Phys. Commun.* **135**, 238 (2001). [arXiv:hep-ph/0010017](#).
- [3] G. Ingelman, A. Edin, and J. Rathsman, *Comput. Phys. Commun.* **101**, 108 (1997). [hep-ph/9605286](#).
- [4] G. Gustafson, *Phys. Lett.* **B175**, 453 (1986).
- [5] G. Gustafson and U. Pettersson, *Nucl. Phys.* **B306**, 746 (1988).
- [6] G. Marchesini and B. R. Webber, *Nucl. Phys.* **B310**, 461 (1988).
- [7] G. Corcella *et al.*, *JHEP* **01**, 010 (2001). [hep-ph/0011363](#).
- [8] B. Andersson, G. Gustafson, and L. Lönnblad, *Nucl. Phys.* **B339**, 393 (1990).
- [9] K. Hamacher and M. Weierstall (1995). [hep-ex/9511011](#).
- [10] V. N. Gribov and L. N. Lipatov, *Yad. Fiz.* **15**, 781 (1972).
- [11] L. N. Lipatov, *Sov. J. Nucl. Phys.* **20**, 94 (1975).
- [12] G. Altarelli and G. Parisi, *Nucl. Phys.* **B126**, 298 (1977).
- [13] Y. L. Dokshitzer, *Sov. Phys. JETP* **46**, 641 (1977).
- [14] B. Andersson, G. Gustafson, L. Lönnblad, and U. Pettersson, *Z. Phys.* **C43**, 625 (1989).
- [15] E. A. Kuraev, L. N. Lipatov, and V. S. Fadin, *Sov. Phys. JETP* **44**, 443 (1976).
- [16] E. A. Kuraev, L. N. Lipatov, and V. S. Fadin, *Sov. Phys. JETP* **45**, 199 (1977).
- [17] I. I. Balitsky and L. N. Lipatov, *Sov. J. Nucl. Phys.* **28**, 822 (1978).
- [18] H1 Collaboration, C. Adloff *et al.*, *Nucl. Phys.* **B538**, 3 (1999). [hep-ex/9809028](#).
- [19] ZEUS Collaboration, J. Breitweg *et al.*, *Eur. Phys. J.* **C6**, 239 (1999). [hep-ex/9805016](#).
- [20] ZEUS Collaboration, S. Chekanov *et al.* (2005). [hep-ex/0502029](#).
- [21] N. Brook, R. G. Waugh, T. Carli, R. Mohr, and M. Sutton. Prepared for Workshop on Future Physics at HERA (Preceded by meetings 25-26 Sep 1995 and 7-9 Feb 1996 at DESY), Hamburg, Germany, 30-31 May 1996.
- [22] L. Lönnblad, *Z. Phys.* **C65**, 285 (1995).

- [23] L. Lönnblad, Nucl. Phys. **B458**, 215 (1996). hep-ph/9508261.
- [24] CDF Collaboration, T. Affolder *et al.*, Phys. Rev. Lett. **84**, 845 (2000). hep-ex/0001021.
- [25] D0 Collaboration, B. Abbott *et al.*, Phys. Rev. **D61**, 032004 (2000). hep-ex/9907009.
- [26] N. Lavesson and L. Lonnblad, JHEP **07**, 054 (2005). hep-ph/0503293.
- [27] E. Thome (2004). hep-ph/0401121.
- [28] E. L. Nurse. FERMILAB-THESIS-2005-05.
- [29] E. Richter-Was and B. Kersevan, *The Monte Carlo Event Generator AcerMC and package AcderDET*. These Proceedings.
- [30] Z. Czynzula *et al.*, *Multi-jet production and Multi-scale QCD*. These Proceedings.
- [31] L. Lönnblad, *THEPEG: Toolkit for high energy physics event generation*. These Proceedings.
- [32] M. Bertini, L. Lönnblad, and T. Sjöstrand, Comput. Phys. Commun. **134**, 365 (2001). hep-ph/0006152.
- [33] S. Catani, F. Krauss, R. Kuhn, and B. R. Webber, JHEP **11**, 063 (2001). hep-ph/0109231.
- [34] L. Lönnblad, JHEP **05**, 046 (2002). hep-ph/0112284.
- [35] S. Höche *et al.*, *Matching Parton Showers and Matrix Elements*. These Proceedings.

The Monte Carlo Event Generator AcerMC and package AcerDET

B. Kersevan^{1,2}, E. Richter-Was^{3,4,*}

¹ Faculty of Mathematics and Physics, University of Ljubljana, Jadranska 19, SI-1000, Slovenia

² Experimental Particle Physics Department, Jozef Stefan Institute, P.P. 3000, Jamova 39, SI-1000 Ljubljana, Slovenia.

³ Institute of Physics, Jagiellonian University, 30-059 Krakow, ul. Reymonta 4, Poland.

⁴ Institute of Nuclear Physics PAN, 31-342 Krakow, ul. Radzikowskiego 152, Poland.

Abstract

The **AcerMC** Monte Carlo Event Generator is dedicated for the generation of Standard Model background processes at pp LHC collisions. The program itself provides a library of the massive matrix elements and phase space modules for generation of selected processes. The hard process event, generated with one of these modules, can be completed by the initial and final state radiation, hadronisation and decays, simulated with either PYTHIA, ARIADNE or HERWIG Monte Carlo event generator and (optionally) with TAUOLA and PHOTOS. Interfaces to all these packages are provided in the distribution version. The matrix element code has been derived with the help of the MADGRAPH package. The phase-space generation is based on the multi-channel self-optimising approach using the modified Kajantie-Byckling formalism for phase space construction and further smoothing of the phase space was obtained by using a modified ac-VEGAS algorithm.

1 Introduction

The physics programme of the general purpose LHC experiments, ATLAS [1] and CMS [2], focuses on the searches for the *New Physics* with the distinctive signatures indicating production of the Higgs boson, SUSY particles, exotic particles, etc. The expected environment will in most cases be very difficult, with the signal to background ratio being quite low, on the level of a few percent after final selection in the signal window.

Efficient and reliable Monte Carlo generators, which enable one to understand and predict background contributions, are becoming key elements in the discovery perspective. As the cross-section for signal events is rather low, even rare Standard Model processes might become the overwhelming backgrounds in such searches. In several cases, generation of such processes is not implemented in the general purpose Monte Carlo generators, when the complicated phase space behaviour requires dedicated (and often rather complex) pre-sampling, whilst the general purpose Monte Carlo generators due to a large number of implemented processes tend to use simpler (albeit more generic) phase space sampling algorithms. In addition, the matrix element expressions for these processes are often quite lengthy and thus require complicated calculations. Only recently, with the appearance of modern techniques for automatic computations, their availability *on demand* became feasible for the tree-type processes. With the computation power becoming more and more easily available even very complicated formulas can now be calculated within a reasonable time frame.

2 The Monte Carlo Event Generator AcerMC

The physics processes implemented in **AcerMC** library [3–5] represent such a set of cases. They are all being key background processes for the discovery in the channels characterised by the presence of the

*Supported in part by the Polish Government grant KBN 1 P03 091 27 (years 2004-2006) and by the EC FP5 Centre of Excellence “COPIRA” under the contract No. IST-2001-37259.

heavy flavour jets and/or multiple isolated leptons. For the Higgs boson searches, the $t\bar{t}H$, ZH , WH with $H \rightarrow b\bar{b}$, the $gg \rightarrow H$ with $H \rightarrow ZZ^* \rightarrow 4\ell$, the $b\bar{b}h/H/A$ with $h/H/A \rightarrow \tau\tau, \mu\mu$ are the most obvious examples of such channels.

Let us shortly discuss the motivation for these few Standard Model background processes which are implemented in the **AcerMC 2.x** library.

The $t\bar{t}b\bar{b}$ production is a dominant irreducible background for the Standard Model (SM) and Minimal Supersymmetric Standard Model (MSSM) Higgs boson search in the associated production, $t\bar{t}H$, followed by the decay $H \rightarrow b\bar{b}$. Proposed analysis [1] requires identifying four b-jets, reconstruction of both top-quarks in the hadronic and leptonic mode and visibility of the peak in the invariant mass distribution of the remaining b-jets. The irreducible $t\bar{t}b\bar{b}$ background contributes about 60-70% of the total background from the $t\bar{t}$ events ($t\bar{t}b\bar{b}$, $t\bar{t}bj$, $t\bar{t}jj$).

The $Wb\bar{b}$ production is recognised as a substantial irreducible background for the Standard Model (SM) and Minimal Supersymmetric Standard Model (MSSM) Higgs boson search in the associated production, WH , followed by the decay $H \rightarrow b\bar{b}$.

The $Wt\bar{t}$ production is of interest because it contributes an overwhelming background [7] for the measurement of the Standard Model Higgs self-couplings at LHC in the most promising channel $pp \rightarrow HH \rightarrow WWWW$.

The $Z/\gamma^*(\rightarrow \ell\ell)b\bar{b}$ production has since several years been recognised as one of the most substantial irreducible (or reducible) backgrounds for the several Standard Model (SM) and Minimal Supersymmetric Standard Model (MSSM) Higgs boson decay modes as well as for observability of the SUSY particles. There is a rather wide spectrum of *regions of interest* for this background. In all cases the leptonic Z/γ^* decay is asked for, but events with di-lepton invariant mass around the mass of the Z-boson mass or with the masses above or below the resonance peak could be of interest. The presented process enters an analysis either by the accompanying b-quarks being tagged as b-jets, or by the presence of leptons from the b-quark semi-leptonic decays in these events, in both cases thus contributing to the respective backgrounds.

The $Z/\gamma^*(\rightarrow \ell\ell, \nu\nu, b\bar{b})t\bar{t}$ production is an irreducible background to the Higgs search in the invisible decay mode (case of $Z \rightarrow \nu\nu$) in the production with association to the top-quark pair [8]. With the $Z/\gamma^*(\rightarrow b\bar{b})$ it is also an irreducible resonant background to the Higgs search in the $t\bar{t}H$ production channel but with the Higgs boson decaying to the b-quark pair.

The complete **EW production** of the $gg, q\bar{q} \rightarrow (Z/W/\gamma^* \rightarrow)b\bar{b}t\bar{t}$ final state is also provided. It can be considered as a benchmark for the previous process, where only the diagrams with resonant $gg, q\bar{q} \rightarrow (Z/\gamma^* \rightarrow)b\bar{b}t\bar{t}$ are included. It thus allows the verification of the question, whether the EW resonant contribution is sufficient in case of studying the $t\bar{t}b\bar{b}$ background away from the Z-boson peak, like for the $t\bar{t}H$ with Higgs-boson mass of 120 GeV.

The $gg, q\bar{q} \rightarrow t\bar{t}t\bar{t}$ production, interesting process per se, is a background to the possible Higgs self-coupling measurement in the $gg \rightarrow HH \rightarrow WWWW$ decay, [7].

The $gg, q\bar{q} \rightarrow (WWbb \rightarrow)f\bar{f}f\bar{f}b\bar{b}$ and $gg, q\bar{q} \rightarrow (t\bar{t} \rightarrow)f\bar{f}b\bar{f}b\bar{b}$ processes give possibility to study spin correlations in the top-quark pair production and decays as well as the effect from the off-shell production. Those are important for the selection optimisation eg. in the $gg \rightarrow H \rightarrow WW$ channel, see the discussion in [9]. As an example, Fig. 1 illustrates spin correlation effects in the top-pair production and decays, namely asymmetry in the correlations between lepton and antilepton direction in the rest frame of top-quark, for events generated with $2 \rightarrow 6$ matrix element. Such correlation is absent if only $2 \rightarrow 2$ matrix element is used for events generation, followed by the independent decays of each top-quark.

A set of **control channels**, i.e. the $q\bar{q} \rightarrow Z/\gamma^* \rightarrow \ell\ell$, $gg, q\bar{q} \rightarrow t\bar{t}$, $q\bar{q} \rightarrow W \rightarrow \ell\nu$ and $gg \rightarrow (t\bar{t} \rightarrow)WbW\bar{b}$ processes, have been added to **AcerMC** in order to provide a means of consistency and cross-check studies.

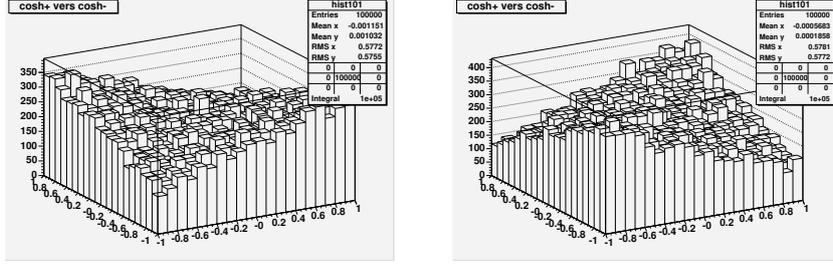


Fig. 1: The correlations between $\cos \Theta$ (azimuthal angle) of lepton and antilepton from $t\bar{t} \rightarrow \ell\bar{\nu}\ell\nu\bar{b}\bar{b}$ decays measured in the rest frame of the top-quark with respect to the anti-top quark direction. Left plot is for $gg \rightarrow (WW\bar{b}\bar{b}) \rightarrow f\bar{f}f\bar{f}b\bar{b}$ process, right plot for $q\bar{q} \rightarrow (WW\bar{b}\bar{b}) \rightarrow f\bar{f}f\bar{f}b\bar{b}$ process.

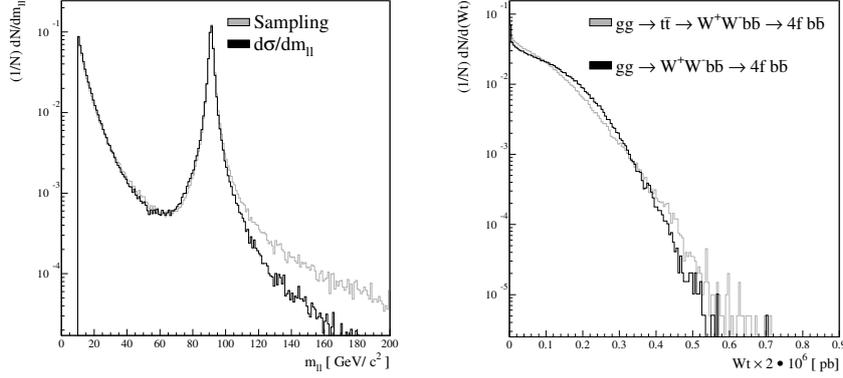


Fig. 2: Left: A representative invariant mass distribution comparisons between the (normalised) sampling functions and the normalised differential cross-section for the $\ell\bar{\ell}$ pair in the process $gg \rightarrow Z^0/\gamma^*b\bar{b} \rightarrow \ell\bar{\ell}b\bar{b}$ process. Right: The weight distribution of the sampled events for the $gg \rightarrow t\bar{t} \rightarrow b\bar{b}W^+W^- \rightarrow b\bar{b}\ell\nu\ell\nu\ell\bar{\nu}\ell\bar{\nu}$ (light gray histogram) and $gg \rightarrow b\bar{b}W^+W^- \rightarrow b\bar{b}\ell\nu\ell\bar{\nu}\ell\bar{\nu}\ell\bar{\nu}$ (black histogram) processes. One can observe the well defined weight range for the two processes; as it turns out the weight distribution is even marginally better for the (more complex) second process, possibly because the higher number of sampling channels manage to cover the event topologies in phase space to a better extent.

This completes the list of the native **AcerMC** processes implemented so far. The hard process event, generated with one of these modules, can be completed by the initial and final state radiation, hadronisation and decays, simulated with either PYTHIA, ARIADNE or HERWIG Monte Carlo event generator and (optionally) with TAUOLA and PHOTOS. Interfaces to all these packages are provided in the distribution releases. The matrix element code has been derived with the help of the MADGRAPH package. The phase-space generation is based on the multi-channel self-optimising approach [3] using the modified Kajantie-Byckling formalism for phase space construction and further smoothing of the phase space was obtained by using a modified ac-VEGAS algorithm.

The improved and automated phase space handling provided the means to include the $2 \rightarrow 6$ processes like e.g. $gg \rightarrow t\bar{t} \rightarrow b\bar{b}W^+W^- \rightarrow b\bar{b}\ell\nu\ell q_1\bar{q}_2$ which would with the very complicated phase space topologies prove to be too much work to be handled manually. The studies show that the overall unweighting efficiency which can be reached in the $2 \rightarrow 6$ processes by using the recommended phase space structuring is on the order of 10 percent. An example of the implemented sampling functions and the actual differential distributions for the $gg \rightarrow Z^0/\gamma^*b\bar{b} \rightarrow \ell\bar{\ell}b\bar{b}$ process and of the weight distribution for the $gg \rightarrow t\bar{t} \rightarrow b\bar{b}W^+W^- \rightarrow b\bar{b}\ell\nu\ell q_1\bar{q}_2$ process are shown in Fig.2

In its latest version, the **AcerMC-2.4** package is interfaced also to ARIADNE 4.1 [12] parton shower model and the LHAPDF structure functions [13].

It is not always the case that the matrix element calculations in the lowest order for a given topology represent the total expected background of a given type. This particularly concerns the heavy flavour content of the event. The heavy flavour in a given event might occur in the hard process of a much simpler topology, as the effect of including higher order QCD corrections (eg. in the shower mechanism). This is the case for the b-quarks present in the inclusive Z-boson or W-boson production, which has a total cross-section orders of magnitude higher than the discussed matrix-element-based $Wb\bar{b}$ or $Zb\bar{b}$ production. Nevertheless, the matrix-element-based calculation is a very good reference point to compare with parton shower approaches in different fragmentation/hadronisation models. It also helps to study matching procedures between calculations in a fixed α_{QCD} order and parton shower approaches. For very exclusive hard topologies matrix-element-based calculations represent a much more conservative approximation than the parton shower ones [6].

3 The AcerDET package

The package **AcerDET** [14] is designed to complete the **AcerMC** generator framework with the easy-to-use simulation and reconstruction algorithms for phenomenological studies on high p_T physics at LHC. The package provides, starting from list of particles in the event, the list of reconstructed jets, isolated electrons, muons and photons and reconstructed missing transverse energy. The **AcerDET** represents a simplified version of the package called ATLFAST [15], used since several years within ATLAS Collaboration. In the **AcerDET** version some functionalities of the former one have been removed, only the most crucial detector effects are implemented and the parametrisations are largely simplified. Therefore it is not representing in details neither ATLAS nor CMS detectors. Nevertheless, we believe that the package can be well adequate for some feasibility studies of the high p_T physics at LHC.

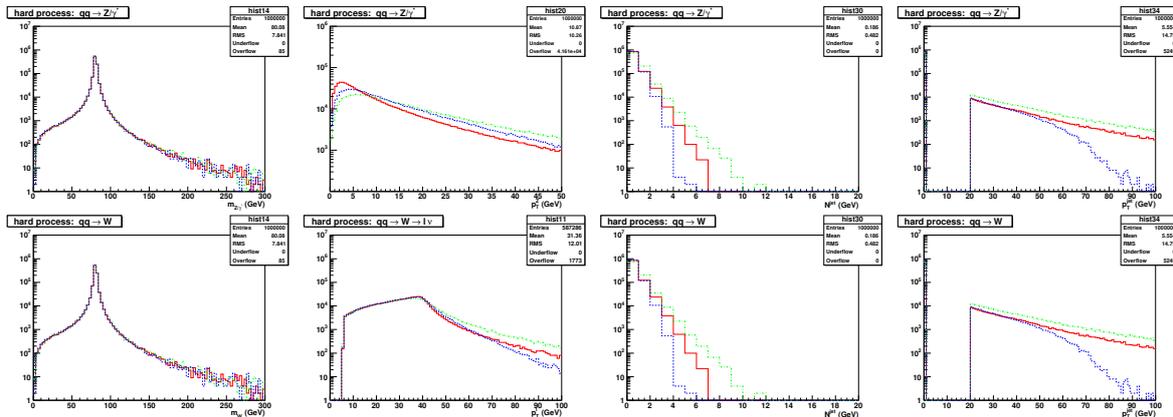


Fig. 3: A few examples of theoretical systematic uncertainties from parton shower model on experimentally observable distributions from Drell-Yan W and Z boson production at LHC (see text).

Fig. 3 shows possible application of the **AcerMC** control processes and **AcerDET** package for studying theoretical systematic uncertainties on the experimentally observed distributions from the parton shower model. The control channels $q\bar{q} \rightarrow Z/\gamma^* \rightarrow \ell\ell$, and $q\bar{q} \rightarrow W \rightarrow \ell\nu$ were processed with parton shower model as implemented in PYTHIA (red), HERWIG (blue) and ARIADNE (green). The comparison includes the distributions of the invariant mass of the di-lepton or lepton-neutrino system, transverse momenta of the Z boson, transverse mass of the W, multiplicity of jets from ISR parton shower and transverse momenta of the hardest jets reconstructed with **AcerDET** package. Perfect agreement on the most left plots confirms consistent starting point for the evolution of the ISR QCD parton shower. The differences observed on remaining plots should be attributed to the systematic theoretical uncertainties of the parton shower models.

References

- [1] ATLAS Collaboration, *ATLAS Detector and Physics Performance TDR*, CERN-LHCC/99-15 (1999).
- [2] CMS Collaboration, Technical Proposal, report CERN/LHCC/94-38 (1994).
- [3] B. Kersevan and E. Richter-Was, *Eur. Phys. J.* **C39** (2005) 439.
- [4] B. Kersevan and E. Richter-Was, *AcerMC version 2.4 with interfaces to PYTHIA 6.2, HERWIG 6.5 and ARIADNE 4.1*, hep-ph/0405247, available from <http://borut.home.cern.ch/borut/>
- [5] B. Kersevan and E. Richter-Was, *Comput. Phys. Commun.* **149** (2003) 142.
- [6] B. Kersevan and E. Richter-Was, *What is the $Wb\bar{b}$ background at LHC?...*, ATLAS Physics Note, ATL-PHYS-2003-018 (2001).
- [7] A. Blondel, A. Clark and F. Mazzucato, ATLAS Physics Note, ATL-PHYS-2002-029 (2002).
- [8] J. Gunion, *Phys. Rev. Lett.* **72** (1994) 199.
- [9] N. Krauer and D. Zeppenfeld, *Phys.Rev.* **D65** (2002) 014021.
- [10] E. Barberio and Z. Was, *Comp. Phys. Commun.* **79** (1994) 291.
- [11] S. Jadach, J. H. Kuhn, Z. Was, *Comput. Phys. Commun.* **64** (1990) 275; M. Jezabek, Z. Was, S. Jadach, J. H. Kuhn, *Comput. Phys. Commun.* **70** (1992) 69; R. Decker, S. Jadach, J. H. Kuhn, Z. Was, *Comput. Phys. Commun.* **76** (1993) 361.
- [12] L. Lönnblad, *Computer Phys. Commun.* **71** (1992) 15.
Manual for ARIADNE version 4 available with the distributed ARIADNE code
- [13] LHAPDF documentation and code available from:
<http://hepforge.cedar.ac.uk/lhapdf/>
- [14] E. Richter-Was, *AcerDET: A particlelevel fast simulation and reconstruction package for phenomenological studies on high $p(T)$ physics at LHC*, hep-ph/0207355.
- [15] E. Richter-Was, D. Froidevaux and L. Poglioli, ATLAS Internal Note ATL-PHYS-98-131 (1998).

RAPGAP

Hannes Jung

Deutsches Elektronen-Synchrotron DESY Hamburg, FRG

Abstract

RAPGAP, originally developed to describe rapidity gap event in ep collisions, has evolved into a multi-purpose Monte Carlo event generator for diffractive and non-diffractive processes at ep colliders both for high Q^2 and in the photoproduction regime ($Q^2 \sim 0$) as well as hard (single diffractive and non-diffractive) processes for pp and $p\bar{p}$ colliders. A detailed description of the program as well as the source code can be found under [1]. In the following only new developments are described.

1 NLO and Order α_s matrix elements

The $\mathcal{O}(\alpha_s)$ matrix elements for light quarks are divergent for $p_T^2 \rightarrow 0$, and usually a p_T^2 cutoff is applied. The \overline{MS} factorization scheme provides a description which finite parts of the matrix elements are treated explicitly and which parts are included in the parton distribution functions. A consistent implementation of the NLO formalism for F_2 in DIS including initial state parton showering is described in detail in [2]. The LO (α_s^0) and the NLO (α_s) part are treated according the \overline{MS} subtraction scheme, reformulated such that it properly can be used together with initial and final state parton showers, avoiding any double counting [3]. When using this scheme, the NLO parton densities calculated in the \overline{MS} scheme should be selected. The program then transforms the parton densities from the \overline{MS} to the BS scheme for parton showers. However, at present only the BGF part is implemented.

2 Les Houches interface

A generic format for the transfer of parton level event configurations from matrix element event generators (MEG) to showering and hadronization event generators (SHG) [4] is provided by the *Les Houches interface*. RAPGAP gives the possibility to write the full parton level events to the file `rapgap.gen`, which can be read in directly by the PYTHIA and HERWIG programs to perform the hadronization. This option is best suited to estimate the uncertainty coming from hadronization correction.

3 Proton dissociation ala DIFFVM

Dissociation of the proton according to the model in DIFFVM [5] can be included for diffractive events. The proton dissociation part of the cross section is given by

$$\frac{d\sigma}{dM_Y^2 dt dx_P} \sim \frac{1}{M_Y^{2(1+\epsilon_Y)}} \exp(-B_{diss}|t|)$$

with ϵ_Y describing the dependence on the dissociation mass M_Y and B_{diss} the t -dependence. The dissociative system Y is split into a *quark – gluon – diquark* system for masses $M_Y > 2$ GeV whereas for masses $0.939 < M_Y < 2$ GeV the system is fragmented according to the nucleon resonances as implemented in DIFFVM [5].

4 Future Plans

In the next future it is planned to include double diffractive scattering for pp collisions to allow simulation of diffractive Higgs production.

References

- [1] H. Jung, *Comp. Phys. Comm.* **86**, 147 (1995);
H. Jung, *The RAPGAP Monte Carlo for Deep Inelastic Scattering, version 3.1*. DESY Hamburg, 2005. <http://www.desy.de/~jung/rapgap/>.
- [2] Collins, John C. and Zu, Xiao-Min, *JHEP* **06**, 018 (2002);
Collins, John C. and Zu, Xiaomin (2004).
- [3] Schilling, S., *Implementation of bgf-processes in monte carlo generators for electron proton scattering*. DESY-THESIS-2000-040.
- [4] Boos, E. and others, *Generic user process interface for event generators*, 2001.
- [5] B. List, *Diffraktive J/ψ Produktion in Elektron - Proton Stößen am Speicherring HERA* (unpublished). Diploma thesis, Techn. Univ. Berlin, H1 note: H1-10/93-319, 1993;
List, B. and Mastroberardino, A., *Diffvm: A monte carlo generator for diffractive processes in ep scattering*, in *Proceedings of the Workshop on Monte Carlo generators for HERA physics*, eds. A. Doyle and G. Grindhammer and G. Ingelman and H. Jung, p. 396. DESY, Hamburg, 1999.

CASCADE

Hannes Jung

Deutsches Elektronen-Synchrotron (DESY), Hamburg, FRG

Abstract

CASCADE is a full hadron-level Monte Carlo event generator for ep , γp , pp and $p\bar{p}$ processes.

CASCADE uses the unintegrated parton distribution functions convoluted with off-mass shell matrix elements for the hard scattering. The CCFM [1] evolution equation is an appropriate description valid for both small and moderate x which describes parton emission in the initial state in an angular ordered region of phase space. For inclusive quantities it is equivalent to the BFKL and DGLAP evolution in the appropriate asymptotic limits. The angular ordering of the CCFM description makes it directly applicable for Monte Carlo implementation. The following processes are available: $\gamma^* g^* \rightarrow q\bar{q}(Q\bar{Q})$, $\gamma g^* \rightarrow J/\psi g$, $g^* g^* \rightarrow q\bar{q}(Q\bar{Q})$ and $g^* g^* \rightarrow h^0$.

A detailed description of CASCADE, the source code and manual can be found under [2]. A discussion of different unintegrated gluon densities can be found in [3–5].

The unintegrated gluon density $x\mathcal{A}_0(x, k_\perp, \bar{q})$ is a function of the longitudinal momentum fraction x the transverse momentum of the gluon k_\perp and the scale (related to the angle of the gluon) \bar{q} . Given this distribution, the generation of a full hadronic event is separated into three steps:

- The hard scattering process is generated,

$$\sigma = \int dk_{t1}^2 dk_{t2}^2 dx_1 dx_2 \mathcal{A}(x_1, k_{t1}, \bar{q}) \mathcal{A}(x_2, k_{t2}, \bar{q}) \sigma(g_1^* g_2^* \rightarrow X), \quad (1)$$

with X being $q\bar{q}$, $Q\bar{Q}$, $J/\psi g$ or h^0 states. The hard cross section is calculated using the off-shell matrix elements given in [6] for $q\bar{q}$ and $Q\bar{Q}$, $\gamma g^* \rightarrow J/\psi g$ in [7] and for Higgs production $g^* G^* \rightarrow h^0$ in [8]. The gluon momentum is given in Sudakov representation:

$$k = x_g p_p + \bar{x}_g p_e + k_t \simeq x_g p_p + k_t. \quad (2)$$

where the last expression comes from the high energy approximation ($x_g \ll 1$), which then gives $-k^2 \simeq k_t^2$.

- The initial state cascade is generated according to CCFM in a backward evolution approach.
- The hadronization is performed using the Lund string fragmentation implemented in PYTHIA [9].

The backward evolution there faces one difficulty: The gluon virtuality enters in the hard scattering process and also influences the kinematics of the produced quarks and therefore the maximum angle allowed for any further emission in the initial state cascade. This virtuality is only known after the whole cascade has been generated, since it depends on the history of the gluon evolution (as \bar{x}_g in eq.(2) may not be neglected for exact kinematics). In the evolution equations itself it does not enter, since there only the longitudinal energy fractions z and the transverse momenta are involved. This problem can only approximately be overcome by using $k^2 = k_t^2/(1 - x_g)$ for the virtuality which is correct in the case of no further gluon emission in the initial state. This problem is further discussed in [5, 10]

The CCFM evolution equations have been solved numerically [11] using a Monte Carlo method. Several sets of un-integrated gluon densities are available which have the input parameters were fitted to describe the structure function $F_2(x, Q^2)$ in the range $x < 5 \cdot 10^{-3}$ and $Q^2 > 4.5 \text{ GeV}^2$ as measured at H1 [12] and ZEUS [13].

Also the unintegrated gluon densities described in [5] including non-linear terms [14] are available within CASCADE.

References

- [1] M. Ciafaloni, Nucl. Phys. **B 296**, 49 (1988);
S. Catani, F. Fiorani and G. Marchesini, Phys. Lett. **B 234**, 339 (1990);
S. Catani, F. Fiorani and G. Marchesini, Nucl. Phys. **B 336**, 18 (1990);
G. Marchesini, Nucl. Phys. **B 445**, 49 (1995).
- [2] H. Jung, Comp. Phys. Comm. **143**, 100 (2002). <http://www.desy.de/~jung/cascade/>.
- [3] Bo Andersson *et al.*, Eur. Phys. J. **C25**, 77 (2002).
- [4] J. R. Andersen *et al.*, Eur. Phys. J. **C35**, 67 (2004).
- [5] J. Collins, M. Diehl, H. Jung, L. Lönnblad, M. Lublinsky and T. Teubner, *Unintegrated parton density functions*. These proceedings, 2005.
- [6] S. Catani, M. Ciafaloni and F. Hautmann, Nucl. Phys. **B 366**, 135 (1991).
- [7] V. A. Saleev and N. P. Zotov, Mod. Phys. Lett. **A 9**, 151 (1994).
- [8] F. Hautmann, Phys. Lett. **B535**, 159 (2002).
- [9] T. Sjostrand *et al.*, Comput. Phys. Commun. **135**, 238 (2001).
- [10] J. Collins and H. Jung, *Need for fully unintegrated parton densities*, 2005.
- [11] H. Jung and G. Salam, Eur. Phys. J. **C 19**, 351 (2001).
- [12] S. Aid *et al.* [H1 Collaboration], Nucl. Phys. **B 470**, 3 (1996);
C. Adloff *et al.* [H1 Collaboration], Eur. Phys. J. **C 21**, 33 (2001).
- [13] M. Derrick *et al.* [ZEUS Collaboration], Z. Phys. **C72**, 399 (1996);
S. Chekanov *et al.* [ZEUS Collaboration], Eur. Phys. J. **C21**, 443 (2001).
- [14] K. Kutak and A. M. Stasto, Eur. Phys. J. **C41**, 343 (2005);
M. Lublinsky, *Parameterization of the dipole cross section and updf*.
<http://www.desy.de/~lublinm/>.

Leading proton production in ep and pp experiments: how well do high-energy physics Monte Carlo generators reproduce the data?

G. Bruni, G. Iacobucci, L. Rinaldi, M. Ruspa
INFN Bologna and University of Eastern Piedmont

Abstract

The simulation of leading-proton production at high-energy colliders as obtained by the HERWIG, LEPTO and PYTHIA Monte Carlo generators is analysed and compared to the measurements of HERA and fixed-target experiments. The discrepancies found between real data and Monte Carlo events could be responsible for inaccurate simulation of particle multiplicities and hadronic final states, which could eventually generate problems in computing the Standard-Model backgrounds to new physics at the LHC collider.

1 Introduction

The production of final state baryons carrying a large fraction of the available energy but a small transverse momentum (leading baryons) is crucial for a deep understanding of strong interactions beyond the perturbative expansion of QCD. Indeed, in high-energy collisions, the QCD-hardness scale decreases from the central, large p_T region, to the soft, non-perturbative hadronic scale of the target-fragmentation region. Therefore, the measurement of leading baryons in the final state of high-energy collisions allows to gather information on the non-perturbative side of strong interactions.

Another reason of interest in leading-baryon production comes from the fact that the energy carried away by the leading baryon(s) produced in a high-energy collision is not available for the production of the central-hadronic system. Therefore, the leading-baryon spectra should be well simulated for a proper accounting of the hadronic multiplicities and energies, e.g. at the LHC collider where an appropriate simulation of these quantities will be the ground for a reliable calculation of the Standard-Model backgrounds to new physics.

Here we will review the data on the production of leading protons and compare them to the most popular Monte Carlo generators available.

2 The data and the Monte Carlo generators used for the comparison

2.1 The proton-proton data

Although the experimental data on leading-proton production are scarce, a few measurements in a large x_L range are available, where x_L represents the fractional longitudinal momentum of the proton. In proton-proton collisions, leading-proton production has been studied both at the ISR [2, 3] and in fixed-target experiments [4–6]. The x_L spectra measured in fixed-target experiments are shown in Fig. 1a-c,e.

2.2 The ep data

Cross sections for the production of leading protons were also measured at the HERA collider [7–9]. More recently, the ZEUS Collaboration made a new measurement [10] of the cross-section for the semi-inclusive reaction $ep \rightarrow eXp$ in deep-inelastic scattering using 12.8 pb^{-1} of data collected during 1997. The single-differential cross sections, $d\sigma_{ep \rightarrow eXp}/dx_L$ and $d\sigma_{ep \rightarrow eXp}/p_T^2$, and the double-differential cross section, $d^2\sigma_{ep \rightarrow eXp}/dx_L dp_T^2$, were measured in the kinematic range $Q^2 > 3 \text{ GeV}^2$ and $45 < W < 225 \text{ GeV}$, where W is the total mass of the hadronic system. The protons were measured using the leading-proton spectrometer (LPS) [11] in the range $x_L > 0.56$ and $p_T^2 < 0.5 \text{ GeV}^2$, where p_T is the scattered-proton transverse momentum.

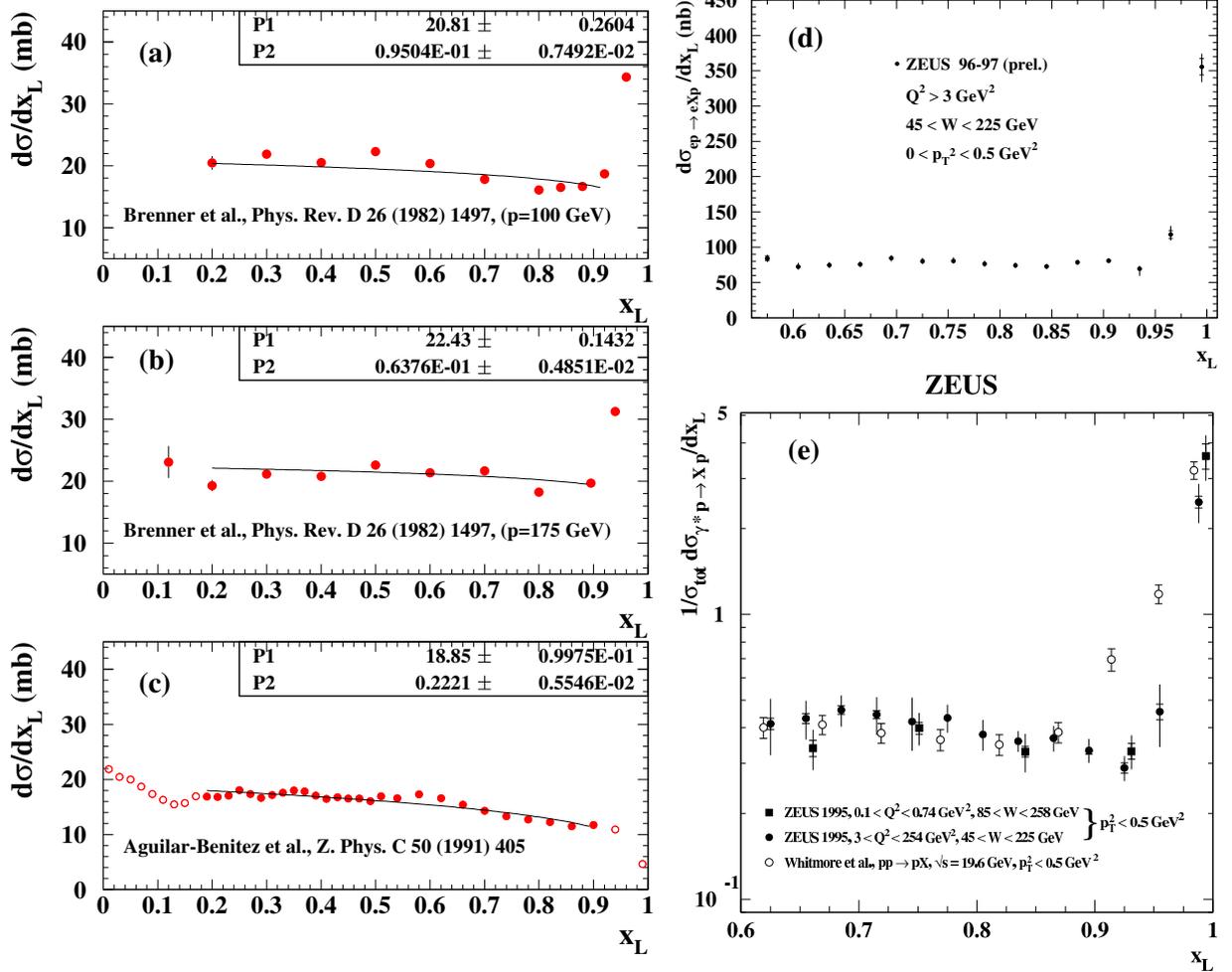


Fig. 1: Differential cross sections $d\sigma/dx_L$ measured in fixed-target experiments (a, b, c and e) and by the ZEUS Collaboration [9, 10] (d, e).

2.3 The Monte Carlo generators

Large samples of Monte Carlo ep events were generated to be compared to the data. The LEPTO generator was used either with the MEPS or the ARIADNE packages; in the latter case the diffractive component of the cross section was simulated using the Soft Color Interaction model. Events were also generated with HERWIG. Since this Monte Carlo does not simulate diffractive events, the POMWIG generator was used to account for the single diffractive events, and the SANG generator to account for the diffractive events in which the scattered-proton dissociates in a higher-mass hadronic system.

Proton-proton events at the LHC center of mass energy (14 TeV) were generated with PYTHIA.

3 Discussion

3.1 The x_L spectrum

Figure 1a, b and c show the $d\sigma/dx_L$ obtained by the fixed target experiments [4–6] which measured leading protons in a wide range of x_L . The cross section for such events shows a peak for values of the final-state proton momentum close to the maximum kinematically allowed value, the so-called diffractive peak. Below the diffractive peak the cross section is lower and consistent with a flat one. In this region, under the assumption of Regge factorisation [12], the fraction of events with a leading proton is expected to be approximately independent of the energy and type of the incoming hadron. The lines superimposed

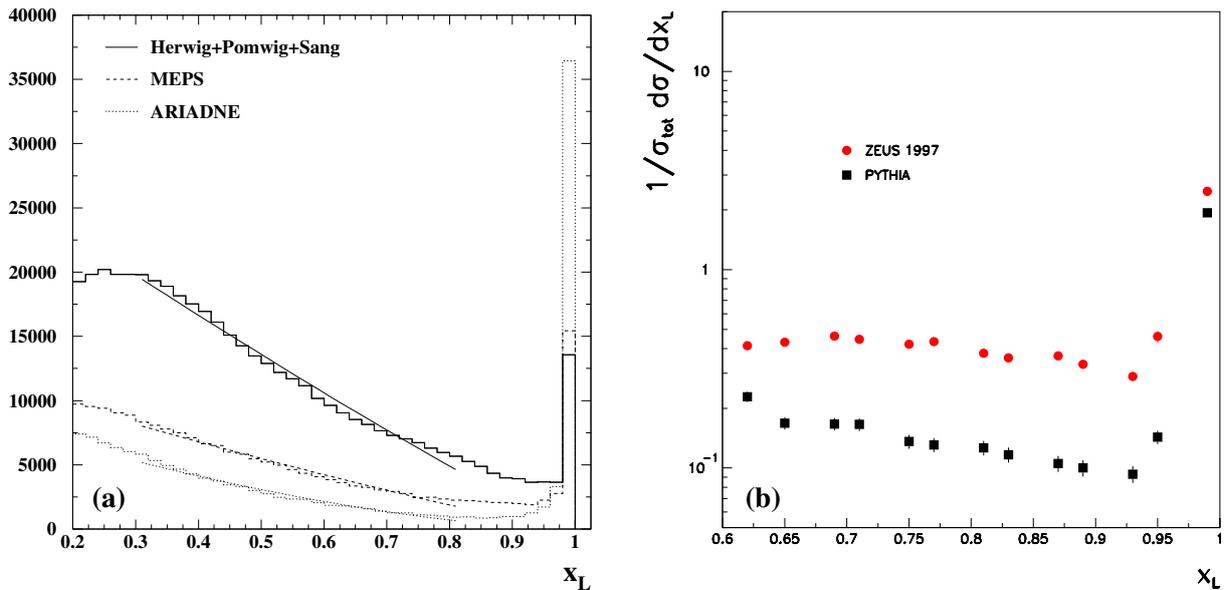


Fig. 2: (a) Differential cross section $d\sigma_{ep \rightarrow eXp}/dx_L$ generated with LEPTO, with MEPS or ARIADNE, and with HERWIG+POMWIG+SANG Monte Carlos; (b) Comparison between the normalised differential cross section $1/\sigma_{tot} d\sigma_{pp \rightarrow Xp}/dx_L$ simulated with PYTHIA and $1/\sigma_{tot} d\sigma_{ep \rightarrow eXp}/dx_L$ measured by the ZEUS Collaboration [9].

to the data are the results of fits in the range $0.1 < x_L < 0.9$ to the function $(1 - x_L)^\alpha$ that is commonly used to characterise the longitudinal distributions of leading particles. The values of α obtained are 0.1, 0.06 and 0.22 respectively for Fig. 1a, b and c. Fig. 1d shows the preliminary $d\sigma/dx_L$ obtained by ZEUS. Below the diffractive peak the cross section is again consistent with a flat one, i.e. $\alpha \sim 0$. A comparison between the normalised cross section $1/\sigma_{tot} d\sigma/dx_L$ obtained by the fixed-target data [6] and by the ep data is shown in Fig. 1e. For $x_L < 0.9$ the fraction of events with a leading proton is indeed consistent for the pp and the ep data set.

The x_L distributions of the simulation of the HERA events are shown in Fig. 2a. Already at first glance, the difference w.r.t. the data is evident, since the spectra are much more populated at low x_L in the Monte Carlo than they are in the data. Indeed, the fits to the same functional form as for the data give $\alpha = 1.0$ for LEPTO-MEPS, $\alpha = 1.4$ for LEPTO-ARIADNE and $\alpha = 1.0$ for HERWIG+POMWIG+SANG.

In Fig. 2b the x_L distribution obtained from the simulation with PYTHIA of pp events at the LHC center of mass energy (14 TeV) is compared to the ZEUS data. As discussed previously, according to the vertex factorisation hypothesis, the fraction of events with a leading proton is expected to be consistent in the ep and in the pp case. The simulation appears to approximately agree with the data in the diffractive peak region but is not able to describe the data neither in shape nor in normalisation in the region outside the diffractive peak.

In general we conclude that the fraction of beam energy carried away by the leading proton in the Monte Carlo is on average much smaller than in the data, with the consequence that the energy available in the simulation for the production of the central-hadronic system is correspondingly larger than in nature.

3.2 The p_T^2 spectrum

Although the p_T^2 distribution of the leading proton is less important for the hadronic final states than the x_L distribution, it is interesting to investigate how well the generators can reproduce it.

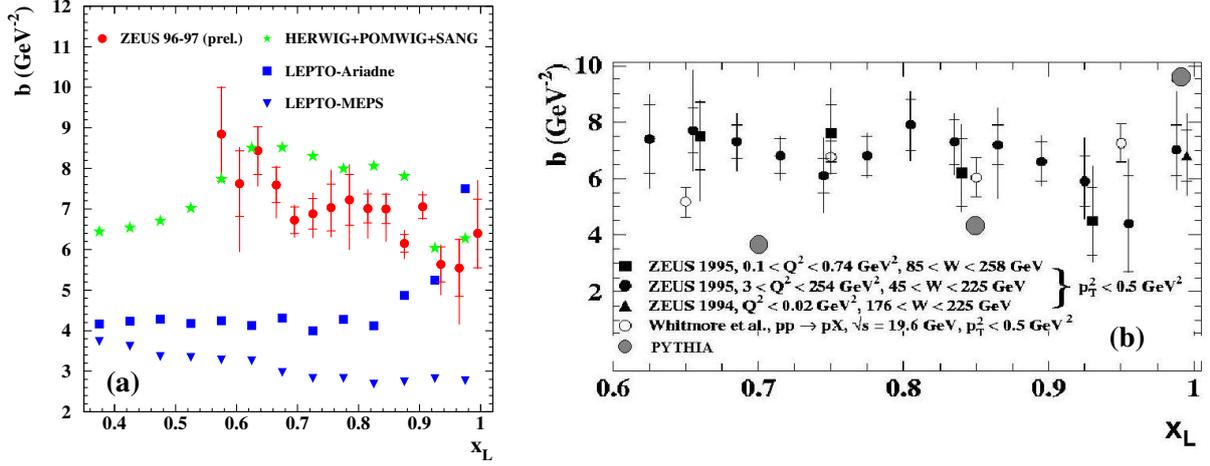


Fig. 3: The slope-parameters b obtained from a single-exponential fit to the differential cross section $d\sigma_{ep \rightarrow eXp}/dp_T^2$ ($d\sigma_{pp \rightarrow Xp}/dp_T^2$ for PYTHIA pp sample) to the function $A \cdot e^{-b \cdot p_T^2}$ in each of the x_L bins shown. (a) The red dots are the ZEUS preliminary data [10], while the other symbols represent the results of the Monte Carlos described in the picture; (b) The grey dots are the pp events simulated with PYTHIA, while the other symbols are the data described in the picture.

In Fig. 3a the red dots show the values of the slope-parameter b obtained from a single-exponential fit to the function $e^{-b p_T^2}$ in each of the x_L bins of the ZEUS $d\sigma/dp_T^2$ measurement. The b slopes obtained by a similar fit performed on the simulated events in the ep case and in the pp case are also reported in Fig. 3a and Fig. 3b, respectively. In the pp case the extracted b -slopes have been corrected for the expected shrinkage of the diffractive peak.

The b -slopes values resulting from the fit to the p_T^2 distribution of the HERWIG+POMWIG+SANG sample appear to be in the right ballpark.

The LEPTO generator shows too small b -slope values, smaller than those of the data by approximately 3 GeV^{-2} . In the case the matrix-element parton showers are used to generate the events, since the dependence of b on x_L is similar to that of the data, it is conceivable to fix the difference by tuning the primordial k_T of the generation. If the ARIADNE package is used instead, it seems quite difficult to improve the situation in a similar way, since the generated b values increase with x_L , a feature that is not seen in the data.

The b -slopes values resulting from the fit to the p_T^2 distribution of the PYTHIA pp sample are approximately consistent with the ZEUS data in the diffractive peak region, but lower than the data again by approximately 3 GeV^{-2} in the region outside the peak.

3.3 Reweighting of the PYTHIA leading proton spectrum

A sample of pp proton events generated with PYTHIA has been used to simulate the many interactions per bunch crossing (pile-up events) occurring at the LHC luminosities in a recent study on the diffractive production of a Higgs boson at the LHC [13]. The simulated leading proton spectrum has been reweighted both in x_L and in p_T^2 with the following function, calculated for each x_L bin of Fig. 2b:

$$f(x_L) = [-37.22 + 135.1 \cdot x_L - 148.5 \cdot x_L^2 + 54.3 \cdot x_L^3] \cdot \frac{e^{-b_{ZEUS} t}}{e^{-b_{PYTHIA} t}} \cdot \frac{b_{ZEUS}}{b_{PYTHIA}},$$

where $b_{ZEUS} = 7 \text{ GeV}^{-2}$ and $b_{PYTHIA} = 4.4 \text{ GeV}^{-2}$. The polynomial form in x_L is the result of a fit to the ratio ZEUS/PYTHIA of the differential cross sections $1/\sigma_{tot} d\sigma/dx_L$ of Fig. 2b; thus $f(x_L)$ provides the number of simulated leading protons to be consistent with the ZEUS data.

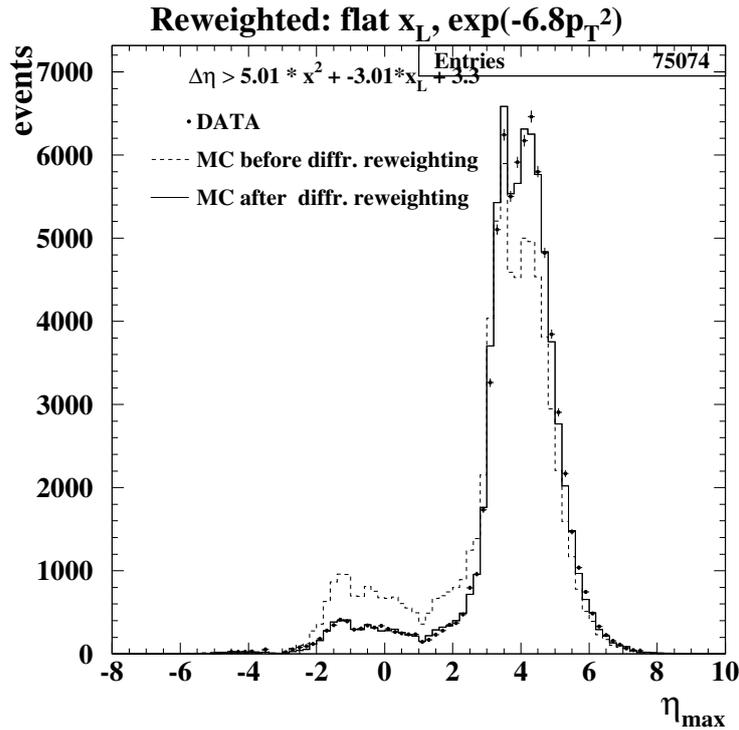


Fig. 4: The η_{max} distribution of the ZEUS data [10] (dots) and of the LEPTO-MEPS Monte Carlo obtained with the default x_L and p_T^2 distributions (dashed-line histogram). The MC distribution after reweighting to a flat x_L and to $e^{-6.8p_T^2}$ is shown by the full-line histogram.

3.4 The fraction of diffractive — large-rapidity gap — events w.r.t. the total

One way to identify a diffractive event produced in ep or pp interactions is to search for a large-rapidity gap (LRG) in the pseudorapidity distribution of the particles produced. In ZEUS, a diffractive-LRG event was tagged by $\eta_{max} < 2$, where η_{max} corresponds to the pseudorapidity of the most forward (i.e. proton direction) energy deposit in the calorimeter exceeding 400 MeV. The η_{max} distribution for ZEUS DIS events with $Q^2 > 3 \text{ GeV}^2$ is shown by the dots in Fig. 4. The two regions of non-diffractive events (with η_{max} between 2 and 8) and of diffractive events (which distribute at η_{max} values below 2) are clearly distinguishable.

The LEPTO-MEPS events were passed through the standard simulation of the ZEUS trigger and detector, and through the same reconstruction and analysis programs as the data. The η_{max} distribution of the MC events after such processing is shown by the dashed histogram in Fig. 4. We note that the diffractive events with $\eta_{max} < 2$ generated with the Soft Color Interaction algorithm in LEPTO-MEPS are more than twice those found in the data. Therefore, the Soft Color Interaction algorithm, as implemented in the LEPTO generator, fails to describe the data in the range of the ZEUS-LPS detector, i.e. $x_L > 0.56$.

4 Summary

The data on leading-proton production in ep and pp scattering have been compared to the most popular Monte Carlo generators available to simulate high-energy physics events. This exercise has revealed that the simulation of the leading-proton momenta, both longitudinal and transverse to the beams, does not reproduce the properties of the data. In particular, the x_L distribution of the leading protons would be made more close to that of the data if a proper accounting of the energy available for the hard-scattering process could be achieved.

Although the HERWIG generator has been successful in simulating many features of high-energy physics final states, it does not contain the diffractive component of the cross section, and the x_L spectrum it produces is far from being almost flat, as seen in the data.

The Soft Color Interaction model in its standard implementation in LEPTO is producing twice the fraction of diffractive-LRG events seen in the ZEUS data at $x_L > 0.56$, therefore distorting in a significant way the multiplicities and hadronic energies present in real events.

The PYTHIA Monte Carlo has been used to simulate pp events at the LHC center of mass energy (14 TeV), and then also compared to ep data. The generator has been shown to reproduce the longitudinal and transverse momentum of the data in the diffractive peak region; however, it underestimates both the cross section and the p_T^2 -slopes at lower values of the scattered proton momentum, contradicting the hypothesis of vertex factorisation, which is supported by the data.

All the above arguments generate some concern that the hadronic multiplicities of the MC generators taken into account here have been tuned consistently and that they can produce an accurate simulation of the final states of the Standard-Model processes at the LHC energies.

5 Acknowledgements

We would like to thank Torbjörn Sjöstrand for useful discussions.

References

- [1] M. Basile et al., Nuovo Cimento 66A (1981) 129.
- [2] M. Basile et al., Lett. Nuovo Cimento 32 (1981) 321.
- [3] V. N. Gribov et al., “The creation of QCD and the effective energy”, World Scientific Series in 20th Century Physics 25 (2001).
- [4] M. Aguilar-Benitez et al., Z. Phys. C 50 (1991) 405.
- [5] A. E. Brenner et al., Phys. Rev. D 26 (1982) 1497.
- [6] J. Withmore et al., Phys. Rev. D 11 (1975) 3124.
- [7] H1 Collab., C. Adloff et al., Nucl. Phys. B 619 (2001) 3.
- [8] H1 Collab., C. Adloff et al., Eur. Phys. J. C 6 (1999) 587.
- [9] ZEUS Collab., S. Chekanov et al., Nucl. Phys. B 658 (2002) 3.
- [10] ZEUS Collab., S. Chekanov et al., Proceedings to the International Europhysics Conference on High Energy Physics (EPS03), Abstract 544.
- [11] ZEUS Collab., M. Derrick et al., Z. Phys. C 3 (1997) 253.
- [12] See e.g.: K. Goulianos and references therein, Phys. Rep. 102 (1983) 169.
- [13] M. Grothe et al., “Diffractive Higgs: CMS/TOTEM Level-1 Trigger Studies”, these proceedings.
- [14] M. Basile et al., Nuovo Cimento 79A (1984) 1.

NLOLIB - A Common Interface for Fixed-Order Calculations

Klaus Rabbertz[†] and Thomas Schörner-Sadenius[°]

[†]University of Karlsruhe, EKP, Postfach 6980, D-76128 Karlsruhe, klaus.rabbertz@cern.ch

[°]University of Hamburg, IExpPh, Luruper Chaussee 149, D-22761 Hamburg, schorner@mail.desy.de

Abstract

We present the current status of the NLOLIB framework which provides an interface to various higher order perturbative calculations, thus allowing for simple comparisons of these calculations with each other and with measured distributions. We show, as a newly included example of the NLOLIB abilities, a comparison of calculations for jet production in deeply inelastic ep scattering.

1 Introduction

Progress in particle physics relies, to a large extent, on the comparison of data to theoretical predictions. Most commonly, the theoretical calculations are available to the experiments in the form of Monte Carlo event generators producing event records that contain all generated particles, their four-momenta, the decay trees etc. in a commonly adopted format. The events that represent the outcome of such a prediction can be used directly by experiment-specific detector simulation and analysis software in order to perform detailed comparisons with experimental data.

Due to complications in the involved mathematical techniques most programs providing higher order electroweak or QCD calculations, however, require large numbers of events to be generated with positive and negative weights. These can not be easily used in simulations because the meaning of the cancellation of positive and negative weights in combination with detector influences is not overly clear. In addition, producing the necessary numbers of events is extremely time-consuming. Nevertheless the results are very useful for comparisons to data and they are used in a variety of experiments to perform, for example, precision measurements of standard model parameters like the strong coupling parameter, α_s .

Although many such programs have been developed for a variety of physical processes, and sometimes even more than one program exist for the same purpose, the usage and the presentation of the results has not been unified so far. This leads to a number of unnecessary technical problems for the user who wants to compare the predictions of more than one program to some measured distribution. To be more specific, the user has to learn, for each single program he or she wants to use, how to install and compile the program, how to implement the specific distributions or processes of interest, and how to extract the results. It should be noted that the programs are also implemented in different programming languages, mostly *FORTRAN* and *C++*.

NLOLIB seeks to simplify the physicists' life by unifying the steering and providing a common framework to implement new quantities and to extract the results. A first version of NLOLIB was developed in the workshop on 'Monte Carlo Generators for HERA Physics' 1998/99 and aimed to integrate and compare three different programs for next-to-leading order calculations in electron-proton scattering [1]. Now this scope will be extended to include also proton-(anti)proton and electron-positron collisions.

2 Implemented programs

RacoonWW provides tree-level cross-sections to all processes where an electron-positron collision yields four fermions in the final state. In addition, it contains a number of higher order corrections, see [2] for details.

DISENT [3] is a next-to-leading order calculation for the production of one or two jets (processes up to $\mathcal{O}(\alpha_s^2)$) for deep-inelastic ep scattering. It uses a subtraction scheme [4] for the cancellation of divergencies. DISENT is the standard program for DIS jet production at HERA and has been used in a variety of analyses.

DISASTER++ [5] offers more possibilities to separate different terms in the derivation of the cross sections; otherwise it provides a functionality similar to DISENT. It also employs the subtraction method.

JetViP [6, 7] is a next-to-leading order calculation for jet production in deep-inelastic ep scattering and e^+e^- collisions that contains processes up to $\mathcal{O}(\alpha_s^2)$ and implements both direct and resolved contributions to the cross-section. The cancellation of divergencies is performed using the phase-space slicing method [8] which leads to dependencies of the resolved contribution on the unphysical slicing parameter, y_{cut} . The direct predictions of JetViP have been shown to be compatible [7, 9] with those of DISENT and DISASTER++. So far, only the implementation into NLOLIB of the direct photon ep scattering part of JetViP has been thoroughly tested; the resolved photon part is implemented in principal but needs more testing. The implementation of the e^+e^- part into NLOLIB has just started.

MEPJET [10] was the first complete next-to-leading order calculation available for deep-inelastic ep scattering and is based on the phase-space slicing method in combination with the technique of crossing functions [11]. The predictions of MEPJET show some discrepancies with respect to the results of DISASTER++, DISENT and JetViP [9].

NLOJET++ [12] incorporates next-to-leading order predictions for ep, pp and $e+e-$ scattering using the subtraction method. Due to its very different way of allowing users to implement their favourite quantities, an integration into NLOLIB on the same footing as for the other programs seems not to be feasible. However, it will be tried to achieve an approach as similar as possible. Currently, only the original version of NLOJET++ in its unchanged form is included.

FMNR [13]: FMNR is a program for the calculation of next-To-leading order photoproduction jet cross-sections with heavy quarks in the final state. The implementation of FMNR into NLOLIB has only just begun.

3 Getting started

Once, a new release is finished, a compressed tar archive will be made available like it is done already now on <http://www.desy.de/~nlolib>.

Since the way in which NLOLIB is installed has changed considerably in the course of this workshop from the original version [1], some short instructions on how to get started are given here.

Originally, the make tool together with a set of perl scripts containing hardware-specific settings were used. However, this procedure was not easily maintainable, so it was decided to employ the GNU autotools [14]. For this to work automake versions 1.7 or higher and autoconf versions 2.57 or higher are needed. In addition, the CERN libraries including a version of PDFLIB [15] and the HzTool [16] libraries as available from our web page are required.

When these conditions are met, the following scheme should be followed for installing NLOLIB:

- Retrieve the NLOLIB source code from the web page at DESY (at a later stage it will also be downloadable from the CVS server in Karlsruhe) and copy it to your working directory (assumed to be `~/nlolib` in the following).
- In `nlolib.sh` (for c-type shells in `nlolib.csh`) set the correct paths to the PDFLIB and the CERNLIB libraries (`libpdflib.a` or `libpdflib804.a`, `libkernlib.a`, `libpacklib.a`), the HzTool libraries (`libhztool.a`, `libmyhztool.a`) and the directories where to put the NLOLIB binaries and libraries, typically `~/nlolib/bin` and `~/nlolib/lib`.
- Go to the working directory and source `nlolib.sh` (or `nlolib.csh`):
`~/nlolib > source nlolib.sh`

- Usually, the following three steps can be skipped. But in case the `configure` script below fails, it has to be recreated by doing:
 - `~ \nlolib > aclocal`
 - `~ \nlolib > automake`
 - `~ \nlolib > autoconf`
- `~ \nlolib > ./configure`
- This step can be skipped if a complete recompile is not necessary/wanted:
 - `~ \nlolib > make clean`
 - `~ \nlolib > make`
 - `~ \nlolib > make install`

Since the running of NLOLIB depends on the simulation program required by the user no general rules can be given here concerning the NLOLIB execution.

4 Jet cross-sections in ep NC DIS

As a new check of the NLOLIB framework we tested the predictions of the DISENT and JetViP programs for deep-inelastic ep scattering single-inclusive jet production against the stand-alone versions of the programs and against data published by the H1 collaboration [17]. The phase-space of the measurement is determined by two requirements on the scattered electron: the energy of the scattered electron E' must be larger than 10 GeV, and its polar angle must be larger than 156° . In addition, two kinematic cuts are applied to select well-reconstructed low- Q^2 DIS events: $5 < Q^2 < 100 \text{ GeV}^2$ and $0.2 < y < 0.6$.

Jet reconstruction for the selected events is performed in the Breit reference frame with the longitudinally invariant k_\perp algorithm [18] in the inclusive mode [19]. Jets are selected by requiring their transverse energy in the Breit frame to be larger than 5 GeV, $E_T^{Breit} > 5 \text{ GeV}$, and their pseudorapidity in the laboratory frame, η^{lab} , to be between -1 and 2.5 .

4.1 Comparison of calculations and data

We first present a comparison of event and jet quantities between the stand-alone versions and the versions implemented in NLOLIB of DISENT and JetViP. For this purpose 1 million events have been generated with each of the four programs using CTEQ4M as proton PDFs and Q^2 as renormalization and factorization scale. Figure 1 shows the differential cross-sections as functions of Q^2 , y , E_e and θ_e for the four predictions. A very good agreement between the predictions is observed.

Also the comparison of the various predictions for the jet pseudorapidities in the Breit and laboratory reference frames, η^{Breit} and η^{lab} , and for the jet transverse energy in the Breit frame, E_T^{Breit} , shows satisfactory agreement. There are, however, small discrepancies between the two JetViP and the two DISENT predictions in the η^{Breit} distribution and a rather large discrepancy between the stand-alone JetViP prediction on the one hand side and the other three calculations on the other hand side for η^{lab} , see Fig. 2.

Figure 3 finally compares the four predictions to the published H1 data which are presented as inclusive jet cross-sections as functions of E_T^{Breit} in different ranges of η^{lab} . Also for these published observables the agreement of the various predictions is reasonable.

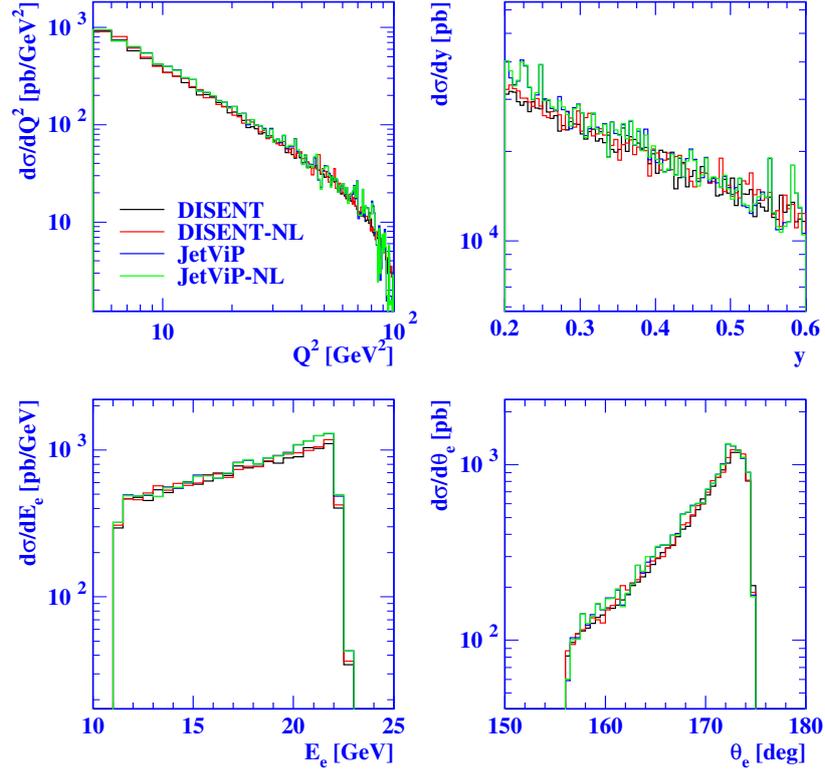


Fig. 1: Cross-sections as functions of Q^2 , y , E_e and θ_e for the different DISENT and JetViP predictions.

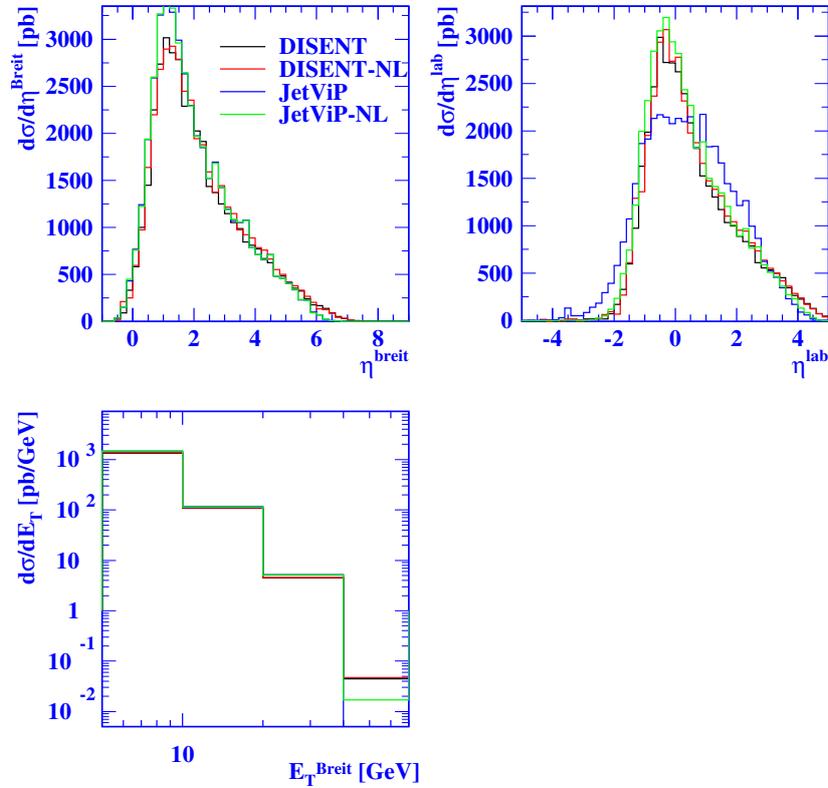


Fig. 2: Inclusive jet cross-sections as functions of η^{Breit} , η^{lab} and E_T^{Breit} for the different DISENT and JetViP predictions.

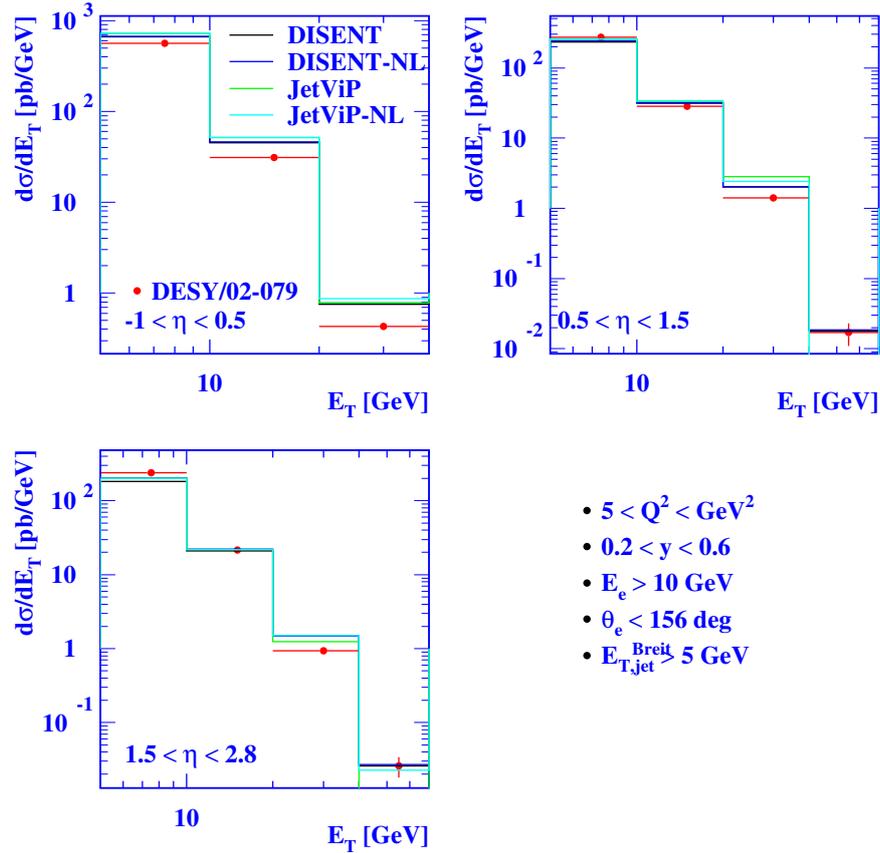


Fig. 3: Inclusive jet cross-sections as functions of E_T^{Breit} in different ranges of η^{lab} . The H1 data are compared to the different DISENT and JetViP predictions.

4.2 How to obtain the theoretical distributions

The NLOLIB calculations shown in this section have been obtained by running DISENT and JetViP via the HzTool interface in NLOLIB and using the HzTool routine for the H1 data analysis, `hz02079.f`. The steering files for the DISENT and JetViP job can be found in `nllib/steering` and are called `dis02079.t` for DISENT and `jv02079.t` for JetViP. The command to run the DISENT job is thus (assuming the command is issued in the `nllib` directory)

```
~ \nllib > bin/hzttol < steering/dis02079.t
```

for JetViP the command is

```
~ \nllib > bin/hzttol < steering/jv02079.t
```

In both cases a HBOOK file `test.hbook` is created that contains, in subdirectory `02079`, the results of the calculation and the published H1 data points. A PAW macro `epjets.kumac` that creates the plots shown here will soon be available.

5 Summary

Some results with recently implemented higher order calculations have been shown, but clearly many items on our agenda unfortunately are still to be done.

For the implementation of JetViP into NLOLIB first, and most importantly, the e^+e^- mode has to be implemented and tested — so far only the ep mode has been done. Secondly, the resolved photon contribution has to be tested more thoroughly and the discrepancies between DISINT and JetViP in the pseudorapidity distributions need to be sorted out.

Concerning NLOJET++ a similar approach like for the other programs has to be set up and thoroughly tested in an all-program comparison, for example of event shapes or jet cross-sections in deep-inelastic scattering. Then, jet cross-sections in hadron-hadron collisions can be derived with NLOJET++. In addition, the use of PDFLIB will be replaced by LHAPDF [20].

Finally, the work on the implementation of more programs, for example FMNR or further proton–(anti)proton programs, needs to be followed up.

References

- [1] T. Hadig and G. McCance, Proceedings of the *Workshop on Monte Carlo Generators for HERA Physics*, eds. G. Grindhammer, G. Ingelman, H. Jung and T. Doyle, DESY, Hamburg (1998), 125.
- [2] A. Denner *et al.*, Phys. Lett. **B475** (2000) 127.
- [3] S. Catani and M. Seymour, Nucl. Phys. **B485** (1997), 291; Erratum *ibid.* **B510** (1998) 503.
- [4] S. Catani and M. Seymour, Proceedings of the Workshop *Future Physics at HERA*, eds. G. Ingelman, A. DeRoeck and R. Klanner, DESY, Hamburg (1996) 519.
- [5] D. Graudenz, arXiv:hep-ph/9710244 (1997).
- [6] B. Pötter, Comput. Phys. Commun. **119** (1999) 45.
- [7] B. Pötter, Comput. Phys. Commun. **133** (2000) 105.
- [8] M. Klasen and G. Kramer, Z. Phys. **C72** (1996)107;
M. Klasen, DESY report DESY-96-204;
M. Klasen, T. Kleinwort and G. Kramer, EPJdirect **C1** (1998) 1.
- [9] C. Duprel *et al.*, Proceedings of the *Workshop on Monte Carlo Generators for HERA Physics*, eds. G. Grindhammer, G. Ingelman, H. Jung and T. Doyle, DESY, Hamburg (1998), 142.
- [10] E. Mirkes and D. Zeppenfeld, Phys. Lett. **B380** (1996) 205;
E. Mirkes, preprint TTP-97-39 (1997), arXiv:hep-ph/9711224.
- [11] W.T. Giele, E.W.N. Glover and D.A. Kosover, Nucl. Phys. **B403** (1993) 633.
- [12] Z. Nagy and Z. Trocsanyi, Phys. Rev. Lett. **87** (2001) 082001.
- [13] S. Frixione *et al.*, Phys. Lett. **B348** (1995) 633;
S. Frixione *et al.*, Nucl. Phys. **B454** (1995) 3.
- [14] <http://www.gnu.org>.
- [15] H. Plochow-Besch, CERN-ETT/TT 2000.04.17 (CERNLIB write-up W5051).
- [16] J. Bromley *et al.*, Proceedings of the Workshop *Future Physics at HERA*, eds. G. Ingelman, A. DeRoeck and R. Klanner, DESY, Hamburg (1996) 611; <http://www.desy.de/~carli/hztool.html>.
- [17] H1 Collaboration, C. Adloff *et al.*, Phys. Lett. **B542** (2002) 193.
- [18] S. Catani *et al.*, Nucl. Phys. **B406** (1993) 187.
- [19] S.D. Ellis and D.E. Soper, Phys. Rev. **D48** (1993) 3160.
- [20] M. Whalley, <http://hepforge.cedar.ac.uk/lhapdf/>.

HZTool

*J. M. Butterworth*¹, *H. Jung*², *V. Lendermann*³, *B. M. Waugh*¹

¹ Department of Physics and Astronomy, University College London, Gower Street, London, WC1E 6BT, England

² Deutsches Elektronen-Synchrotron, Notkestr. 85, 22607 Hamburg, Germany;

³ Kirchhoff-Institut für Physik, Universität Heidelberg, Im Neuenheimer Feld 227, 69120 Heidelberg, Germany

Abstract

HZTOOL is a library of Fortran routines used for tuning and validating of Monte Carlo and analytical models of high energy particle collisions. The library includes an extensive collection of routines reproducing ep , γp , $p\bar{p}$ and $\gamma\gamma$ data published by the HEP experiments, as well as calculations as they were performed in the experimental data analyses. During this workshop the library was further developed to include HERA and TeVatron results relevant for the studies at the LHC.

1 Introduction

Data from high-energy physics experiments have seen the triumph of the Standard Model both in precision electroweak measurements and in the verification of QCD to a reasonable degree of precision. However, a number of aspects of high energy collisions remain poorly understood due to technical difficulties in the calculation. This is particularly the case for measurements of the hadronic final state in high energy collisions, where the specific event shapes variables, jet algorithms and kinematic cuts may be rather complex.

Accurate models of the final state are often needed to design new experiments and to interpret the data from them. Simulation programs employing fits to existing data address these problems. However, consistent tuning of the parameters of these programs, and examination of the physics assumptions they contain, is non-trivial due to the wide variety of colliding beams, regions of phase space, and complex observables. Comparing a new calculation to a sensible set of relevant data is in practice extremely time consuming and prone to error.

HZTOOL [1] is created to improve this situation. It is a library of Fortran routines allowing reproduction of the experimental distributions and easy access to the published data. Basically, each subroutine corresponds to a published paper. If supplied with the final states of a set of simulated collisions, these routines will perform the analysis of the final state exactly as it was performed in the paper, providing simulated data points which may be compared to the measurement. HZTOOL currently contains measurements from ep , γp , $\gamma\gamma$ and $p\bar{p}$ collisions. Others may easily be added.

While it is designed to be used as simply as possible as a standalone library, HZTOOL is also a key component of JETWEB [2]. JETWEB is a Web-based facility for tuning and validating Monte Carlo models. A relational database of reaction data and predictions from different models is accessed via the Web, enabling a user to find out how well a given model agrees with a range of data and to request predictions using parameter sets that are not yet in the database.

HZTOOL can also be used together with RUNMC [3], which is an object-oriented front-end for Monte Carlo programs incorporating a sophisticated graphical user interface.

The library was initially developed within the workshop “Future Physics at HERA” [4] but has since expanded to become a more general toolkit. The package was managed by *T. Carli* for quite some time and many people have contributed routines and general development to HZTOOL since it first appeared. Nowadays, HZTOOL and JETWEB are further developed within the CEDAR project [5]. The

current maintainers are *J. Butterworth, H. Jung, E. Nurse* and *B. Waugh*¹. It is planned to design a C++ equivalent, in order to provide a native interface to the new C++ Monte Carlo programs, to enable a straightforward implementation of new HEP data analyses performed in C++.²

2 HZTOOL Usage

Each analysis subroutine books, fills and outputs two sets of histograms: one reproducing the published data and another one filled by the chosen simulation program. The routine names relate to the publication. The preferred convention is:

`HZHymmnnn` where `yymmnnn` is the arXiv:hep-ex preprint number.

Alternative naming schemes, used for older routines or when a hep-ex number is not available, are:

`HZDyynn` where `yynn` is the DESY preprint number.

`HZCyynn` where `yynn` is the CERN preprint number.

`HZFyynnE` where `yynn` is the FNAL preprint number.

`HZyynn` where `yynn` is the DESY preprint number³.

If none of the above number schemes exist, the routine name is generally derived from the journal publication (e.g. `HZPRT154247`). Occasionally a single publication contains results taken under more than one set of beam conditions, in which case there will be a routine for each beam condition, distinguished by appending a letter to the expected name (e.g. `HZC88172A`, `HZC88172B`).

HZTOOL is a library, and the main program, which is usually a Monte Carlo event generator, must be provided by the user. The code of HZTOOL routines is basically independent of the program used to simulate the collisions. The Monte Carlo generators currently supported are: ARIADNE [6], CASCADE [7], HERWIG [8] (including JIMMY [9]), LEPTO [10], PYTHIA [11], PHOJET [12], QCDINS [13], RAPGAP [14], RIDI [15], and DJANGO [19]. The production versions of these programs are all currently written in Fortran. The library can also be accessed within the NLOLIB framework for running NLO QCD programs [16]. Besides the HZTOOL library it is also necessary to link in the CERMLIB library routines and possibly PDFLIB [17] or LHAPDF [18]. Examples of the main programs can be found in the HZSTEER package [20] which provides the executable programs for JETWEB to submit from its backend.

To ease the implementation of the analysis code, HZTOOL provides the relevant jet finders (the cluster and cone algorithms with various options), as well as a number of utilities to calculate event shape variables, to perform Lorentz boosts etc.

3 Recent Developments

Within this workshop, an effort was made to include all results from HERA and other HEP experiments which can be helpful for tuning of MC models used for event simulations at the LHC. In particular, the models for multiple interactions and for heavy flavour production were considered.

The publications which may be relevant for the tuning of multiple interaction models and which are available in HZTOOL are listed in [21]. The names for the corresponding HZTOOL subroutines are also specified. From this list, the newly written routines are: `HZH9505001` (*J. M. Butterworth, B. M. Waugh*), `HZH9810020` (*S. Lausberg, V. Lendermann*), `HZH0006017` (*D. Beneckenstein, V. Lendermann*) and `HZH0302034` (*K. Lohwasser, V. Lendermann*). The routine `HZ95219` was extended by *A. Buniatian* to include the results from the corresponding H1 paper which are especially sensitive to underlying events in the photoproduction of jets. The models of multiple interactions and efforts of their tuning are reviewed in [22].

¹The maintainers can be contacted at `hztool@cedar.ac.uk`. To receive announcements of new releases, send an e-mail to `majordomo@cedar.ac.uk` with `subscribe hztool-announce` in the body of the e-mail.

²More details may be found at `http://hepforge.cedar.ac.uk/rivet`.

³This naming should not be used for new routines; the HZD prefix is preferred.

As for heavy flavour production, a number of HERA measurements of open charm and beauty production are included in the library [23]. From those, the newly written routines are: HZH0108047 (*P. D. Thompson*), HZH0312057 (*O. Gutsche*), HZH0408149 (*A. W. Jung*). New publications [24] are to be implemented. The following routines for the TeVatron results [25] were also recently provided: HZH9905024 (*O. Gutsche*), HZH0307080 (*H. Jung*), HZH0412071 (*H. Jung, K. Peters*). Furthermore a set of *Benchmark* cross sections have been defined for easy comparison of different calculations: HZDISCC, HZDISBB for charm and beauty production in DIS, HZHERAC, HZHERAB for photoproduction of charm and beauty and HZLHCC, HZLHCBB for charm and beauty production at the LHC [26].

References

- [1] HZTOOL package, manual and tutorial can be downloaded from
<http://hepforge.cedar.ac.uk/hztool/>
- [2] J. M. Butterworth and S. Butterworth, *Comput. Phys. Commun.* **153**, 164 (2003);
<http://jetweb.cedar.ac.uk>
- [3] S. Chekanov, these proceedings, working group 5; <http://hepforge.cedar.ac.uk/runmc/>
- [4] J. Bromley *et al.*, Proceedings of the Workshop “Future physics at HERA”, Hamburg 1995/96, vol. 1, 611-612.
- [5] J. M. Butterworth, S. Butterworth, B. M. Waugh, W. J. Stirling and M. R. Whalley, hep-ph/0412139;
A. Buckley *et al.*, these proceedings, working group 5; <http://www.cedar.ac.uk/>
- [6] L. Lönnblad, *Comput. Phys. Commun.* **71**, 15 (1992);
<http://www.thep.lu.se/~leif/ariadne/>
- [7] H. Jung, G. Salam, *Eur. Phys. J. C* **19**, 351 (2001);
H. Jung, *Comput. Phys. Commun.* **143**, 100 (2002); <http://www.desy.de/~jung/cascade/>
- [8] G. Corcella, I. G. Knowles, G. Marchesini, S. Moretti, K. Odagiri, P. Richardson, M. H. Seymour and B. R. Webber, *JHEP* **0101**, 010 (2001);
<http://hepwww.rl.ac.uk/theory/seymour/herwig>
- [9] J. M. Butterworth, J. R. Forshaw and M. H. Seymour, *Z. Phys. C* **72**, 637 (1996);
<http://hepforge.cedar.ac.uk/jimmy>
- [10] G. Ingelman, A. Edin and J. Rathsman, *Comput. Phys. Commun.* **101**, 108 (1997);
<http://www3.tsl.uu.se/thep/lepto/>
- [11] T. Sjöstrand, P. Eden, C. Friberg, L. Lönnblad, G. Miu, S. Mrenna and E. Norrbin,
Comput. Phys. Commun. **135**, 238 (2001); <http://thep.lu.se/tf2/staff/torbjorn/>
- [12] R. Engel, *Z. Phys. C* **66**, 203 (1995); <http://www.physik.uni-leipzig.de/~engel>
- [13] A. Ringwald and F. Schrempf, *Comput. Phys. Commun.* **132**, 267 (2000);
<http://www.desy.de/~t00fri/qcdins/qcdins.html>
- [14] H. Jung, *Comput. Phys. Commun.* **86**, 147 (1995); <http://www.desy.de/~jung/rapgap/>
- [15] M. G. Ryskin and A. Solano, <http://www-zeus.desy.de/~solano/RIDI2.0/>
- [16] K. Rabbertz and T. Schörner-Sadenius, these proceedings, working group 5;
<http://www.desy.de/~nlolib/>
- [17] H. Plochow-Besch, *Comput. Phys. Commun.* **75**, 396 (1993).
- [18] W. Giele *et al.*, Proceedings of Workshop on Physics at TeV Colliders, Les Houches, France, 2001, [hep-ph/0204316];
M. Whalley, D. Bourilkov and R. C. Group, these proceedings, working group 5;
<http://hepforge.cedar.ac.uk/lhapdf/>
- [19] G. A. Schuler and H. Spiesberger,
<http://wwwthep.physik.uni-mainz.de/~hspiesb/djangoh/djangoh.html>
- [20] HZSTEER package can be downloaded from
<http://hepforge.cedar.ac.uk/hzsteer/>

- [21] M. Derrick *et al.* [ZEUS Collaboration], Phys. Lett. B **342**, 417 (1995), HZ94176;
M. Derrick *et al.* [ZEUS Collaboration], Phys. Lett. B **348**, 665 (1995), HZ95033;
M. Derrick *et al.* [ZEUS Collaboration], Phys. Lett. B **354**, 163 (1995), HZH9505001;
M. Derrick *et al.* [ZEUS Collaboration], Phys. Lett. B **369**, 55 (1996), HZ95194;
S. Aid *et al.* [H1 Collaboration], Z. Phys. C **70**, 17 (1996), HZ95219;
M. Derrick *et al.* [ZEUS Collaboration], Phys. Lett. B **384**, 401 (1996), HZ96094;
K. Ackerstaff *et al.* [OPAL Collaboration], Z. Phys. C **73**, 433 (1997), HZC96132;
C. Adloff *et al.* [H1 Collaboration], Eur. Phys. J. C **1**, 97 (1998), HZ97164;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **2**, 61 (1998), HZ97191;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **1**, 109 (1998), HZ97196;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **4**, 591 (1998), HZ98018;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **6**, 67 (1999), HZ98085;
J. Breitweg *et al.* [ZEUS Collaboration], Phys. Lett. B **443**, 394 (1998), HZ98162;
C. Adloff *et al.* [H1 Collaboration], Eur. Phys. J. C **10**, 363 (1999), HZH9810020;
G. Abbiendi *et al.* [OPAL Collaboration], Eur. Phys. J. C **10**, 547 (1999), HZC98113;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **11**, 35 (1999), HZ99057;
C. Adloff *et al.* [H1 Collaboration], Phys. Lett. B **483**, 36 (2000), HZ00035;
C. Adloff *et al.* [H1 Collaboration], Eur. Phys. J. C **18**, 293 (2000) HZH0006017;
S. Chekanov *et al.* [ZEUS Collaboration], Eur. Phys. J. C **23**, 615 (2002), HZ01220;
C. Adloff *et al.* [H1 Collaboration], Eur. Phys. J. C **25**, 13 (2002), HZ01225;
T. Affolder *et al.* [CDF Collaboration], Phys. Rev. D **65**, 092002 (2002), HZF01211E;
C. Adloff *et al.* [H1 Collaboration], Eur. Phys. J. C **29**, 497 (2003), HZH0302034;
S. Chekanov *et al.* [ZEUS Collaboration], Eur. Phys. J. C **35**, 487 (2004), HZH0404033.
- [22] C. Buttar *et al.*, *Underlying events*, these proceedings, working group 2.
- [23] C. Adloff *et al.* [H1 Collaboration], Z. Phys. C **72**, 593 (1996), HZ96138;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **6**, 67 (1999), HZ98085, HZ98085p;
C. Adloff *et al.* [H1 Collaboration], Nucl. Phys. B **545**, 21 (1999), HZ98204;
C. Adloff *et al.* [H1 Collaboration], Phys. Lett. B **467**, 156 (1999)
[Erratum-ibid. B **518**, 331 (2001)], HZ99126;
J. Breitweg *et al.* [ZEUS Collaboration], Eur. Phys. J. C **18**, 625 (2001), HZ00166;
C. Adloff *et al.* [H1 Collaboration], Phys. Lett. B **528**, 199 (2002), HZ01100;
C. Adloff *et al.* [H1 Collaboration], Phys. Lett. B **520**, 191 (2001), HZH0108047;
S. Chekanov *et al.* [ZEUS Collaboration], Phys. Lett. B **565**, 87 (2003), HZ03015;
S. Chekanov *et al.* [ZEUS Collaboration], Nucl. Phys. B **672**, 3 (2003), HZ03094;
S. Chekanov *et al.* [ZEUS Collaboration], hep-ex/0312057, HZH0312057;
A. Aktas *et al.* [H1 Collaboration], Eur. Phys. J. C **38**, 447 (2005), HZH0408149.
- [24] S. Chekanov *et al.* [ZEUS Collaboration], Phys. Rev. D **69**, 012004 (2004);
S. Chekanov *et al.* [ZEUS Collaboration], Phys. Lett. B **599**, 173 (2004);
A. Aktas *et al.* [H1 Collaboration], Eur. Phys. J. C **40**, 349 (2005);
A. Aktas *et al.* [H1 Collaboration], Eur. Phys. J. C **41**, 453 (2005);
A. Aktas *et al.* [H1 Collaboration], Phys. Lett. B **621**, 56 (2005);
A. Aktas *et al.* [H1 Collaboration], Submitted to Eur. Phys. J. C , hep-ex/0507081;
S. Chekanov *et al.* [ZEUS Collaboration], hep-ex/0508019.
- [25] B. Abbott *et al.* [D0 Collaboration], Phys. Lett. B **487**, 264 (2000), HZH9905024;
D. Acosta *et al.* [CDF Collaboration], Phys. Rev. Lett. **91**, 241804 (2003), HZH0307080;
D. Acosta *et al.* [CDF Collaboration], Phys. Rev. D **71**, 032001 (2005), HZH0412071.
- [26] O. Behnke, M. Cacciari, M. Corradi, A. Dainese, H. Jung, E. Laenen, I. Schienbein and H. Spiesberger, these proceedings, working group 3.

CEDAR

*A Buckley*¹, *J M Butterworth*², *S Butterworth*², *L Lönnblad*³, *W J Stirling*¹, *M Whalley*¹, *B M Waugh*²

¹ Institute for Particle Physics Phenomenology, University of Durham, DH1 3LE, UK

² Department of Physics and Astronomy, UCL, Gower Street, London, WC1E 6BT

³ Department of Theoretical Physics, Sölvegatan 14A, S-223 62 Lund, Sweden

Abstract

The CEDAR collaboration is developing a set of tools for tuning and validating theoretical models by comparing the predictions of event generators with data from particle physics experiments. CEDAR is also constructing resources to provide access to well defined versions of high-energy physics software and support for software developers. Here we give an overview of the CEDAR project and its status and plans.

1 Introduction

Despite the success of the Standard Model in accurately describing a wide range of phenomena in high-energy particle physics, there are aspects of high-energy collisions where technical difficulties in the relevant calculations make it hard to attain a good understanding. This is particularly true where non-perturbative QCD is involved, as in the description of hadronic collisions, where the final state is influenced by the parton distribution functions (PDFs) of the colliding beams, by multiple soft interactions leading to an “underlying event” and by hadronisation of the outgoing partons.

These theoretical uncertainties can limit the precision of new measurements, as well as hindering the planning of future experiments. Building accurate models of hadronic processes is important for these reasons as well as for the insight they may offer into the fundamental physics involved. However, the models that are constructed typically have a number of parameters that can be varied, constrained only by how well the resulting predictions agree with experiment.

Tuning these free parameters and testing the models against experimental data is a difficult task because the data are so varied, involving different beam particles, different regions of phase space and complex observables. Changing a single parameter in a model can affect the predictions for different measurements in very different ways, and tuning to a limited set of data may result in a contradiction with other data not taken into account.

It is thus important to compare models simultaneously with as wide a range as possible of experimental results, and the aim of CEDAR [1] is to simplify this task.

The rest of this contribution will describe in turn the projects making up CEDAR.

2 HZTool and Rivet

The first requirement is for a library of routines to enable, for each experimental measurement of interest, a comparable prediction to be produced from any given Monte Carlo generator. This role is currently filled by the Fortran library HZTool, described in more detail in another contribution to these proceedings [2]. The HZTool library is being maintained by CEDAR, with subroutines for various measurements contributed by a number of authors within and outside the CEDAR collaboration. A number of HERA routines were written within this workshop.

Work is underway to build a replacement for HZTool, to be called Rivet (Robust Independent Validation of Experiment and Theory). This will use an object-oriented design, implemented in C++, together with standard interfaces (such as HepMC [3] and AIDA [4]) to make the new framework more flexible and extensible than the Fortran HZTool. For example, it will be easier to incorporate new Monte Carlo generators into Rivet than into HZTool.

3 JetWeb

JetWeb [5] provides a web interface to HZTool, along with a relational database of both experimental data and model predictions generated using HZTool. The core of JetWeb is a set of Java servlets that manipulate an object model representing data and predictions. A user can use a web form to specify a model and choice of parameters, and the data they wish to compare to this model. If this model and parameter set are already in the database, a set of comparison plots and statistics is returned. Otherwise the user may request a set of Monte Carlo jobs to be run using their specified model, and the results will be added to the database.

The existing JetWeb database has been frozen, although it can still be searched, while the design and functionality are improved. JetWeb is being adapted to use the data already stored in HEPDATA rather than duplicating this in its own database, and to make the addition of further Monte Carlo models (beyond the currently supported HERWIG [6] and PYTHIA [7]) easier.

4 HEPDATA

HEPDATA [8] is a well established and widely used source of scattering data from HEP experiments. As part of CEDAR it has been converted from the existing hierarchical structure to a relational database using MySQL. The next steps in this part of the CEDAR project will provide front ends so that the data in the relational database can be accessed through a searchable web interface and also directly by JetWeb and other users.

5 HepML

In order to simplify the transfer of data between different parts of the CEDAR project and other software frameworks, an XML schema [9] is being developed to specify particle reactions and experimental results (as provided by HEPDATA) as well as generator programs and parameters.

This schema is separate from the HepML developed within the MCDB project [10], which is designed primarily as a format for event records. It may be that some parts of the schemas can be unified later or become parts of a more general schema.

6 HepCode

HepCode [11] aims to provide access to well defined versions of Monte Carlo programs, parton distribution functions and other high-energy physics calculation programs. Currently it is simply a list of codes with details for each of the processes calculated, the order of the calculation, the authors and the programming language used, along with a link to further information where this is available.

HepCode will eventually feature a search facility so that users can find a set of available programs simply by entering the details of a particular scattering process. It may also be possible to have links from matching data records in HEPDATA or from papers in bibliographic databases such as SPIRES.

7 HepForge

CEDAR also provides a development environment, HepForge, for authors of HEP software, including Monte Carlo generators. In addition to the core CEDAR projects (HZTool, HZSteer, JetWeb, HepML) other projects using HepForge are fastNLO [12], Herwig++ [13], Jimmy [14], KtJet [15], LHAPDF [16], RunMC [17] and ThePEG [18].

Facilities provided to developers include a code repository (using CVS or Subversion), a bug tracker (using Trac), a wiki for documentation and communication between project contributors, and mailing lists for project discussions, queries and announcements.

References

- [1] J. M. Butterworth, S. Butterworth, B. M. Waugh, W. J. Stirling and M. R. Whalley, hep-ph/0412139, presented at CHEP'04, Interlaken, September 2004;
<http://www.cedar.ac.uk/>
- [2] J. M. Butterworth, H. Jung, V. Lendermann and B. M. Waugh, *HZTool*, these proceedings, working group 5;
HZTool package, manual and tutorial can be downloaded from
<http://hepforge.cedar.ac.uk/hztool/>
- [3] M. Dobbs and J. B. Hansen, *Comput. Phys. Commun.* **134** 41 (2001);
<http://mdobbs.home.cern.ch/mdobbs/HepMC/>
- [4] <http://aida.freehep.org/>
- [5] J. M. Butterworth and S. Butterworth, *Comput. Phys. Commun.* **153**, 164 (2003);
<http://jetweb.cedar.ac.uk/>
- [6] G. Corcella, I. G. Knowles, G. Marchesini, S. Moretti, K. Odagiri, P. Richardson, M. H. Seymour and B. R. Webber, *JHEP* **0101**, 010 (2001);
<http://hepwww.rl.ac.uk/theory/seymour/herwig/>
- [7] T. Sjöstrand, P. Edén, C. Friberg, L. Lönnblad, G. Miu, S. Mrenna and E. Norrbin, *Comput. Phys. Commun.* **135**, 238 (2001);
<http://www.thep.lu.se/~torbjorn/Pythia.html>
- [8] <http://durpdg.dur.ac.uk/hepdata/>
- [9] <http://hepforge.cedar.ac.uk/hepml/>
- [10] P. Bartalini, L. Dudko, A. Kryukov, I. Seluzhenkov, A. Sherstnev, A. Vologdin, hep-ph/0404241.
- [11] <http://www.cedar.ac.uk/hepcode/>
- [12] <http://hepforge.cedar.ac.uk/fastnlo/>
- [13] <http://hepforge.cedar.ac.uk/herwig/>
- [14] J. M. Butterworth, J. R. Forshaw and M. H. Seymour, *Z. Phys. C* **72**, 637 (1996);
<http://hepforge.cedar.ac.uk/jimmy/>
- [15] J. M. Butterworth, J. P. Couchman, B. E. Cox and B. M. Waugh, *Comp. Phys. Comm.* **153**, 85 (2003)
<http://hepforge.cedar.ac.uk/ktjet/>
- [16] M. Whalley and S. Bourikov, these proceedings, working group 5;
<http://hepforge.cedar.ac.uk/lhapdf/>
- [17] S Chekanov, these proceedings, working group 5;
<http://hepforge.cedar.ac.uk/runmc/>
- [18] L. Lönnblad, these proceedings, working group 5;
<http://hepforge.cedar.ac.uk/thepeg/>

RunMC: an object-oriented analysis framework to generate Monte Carlo events for current and future HEP experiments

S. Chekanov

HEP division, Argonne National Laboratory, 9700 S.Cass Avenue, Argonne, IL 60439, USA

E-mail: chekanov@mail.desy.de

Abstract

RunMC is a C++ object-oriented framework aimed to generate and to analyze high-energy collisions using Monte Carlo simulations. The package provides a common interface to different Monte Carlo models using modern physics libraries developed for the LHC and NLC experiments. Physics calculations can easily be loaded and saved as external project modules. This simplifies the development of complicated physics calculations for high-energy physics in large collaborations.

1 Introduction

Monte Carlo models (MC) written in FORTRAN77 are widely used in many high-energy physics laboratories worldwide. These models are known to be fast, robust and well tested. However, the main choice for future high-energy experiments is an object-oriented programming language, either C++ (the LHC experiments at CERN) or Java (the NLC project). Some steps towards converting the FORTRAN MC models to the C++ programming language have already been undertaken [1]. However, such models written in C++ will require a thoughtful verification to insure that their predictions are consistent with the original FORTRAN-based MC programs, as well as with the physics results obtained in the past. Such verifications will go over certain time, and a tool which allows to perform such comparisons is urgently needed.

A program which allows running of both FORTRAN-coded and C++ MC models using a common C++ programming environment should be valuable. This is important not only for comparisons and verifications of these MC models. Such a C++ framework can also extend the lifetime of FORTRAN-based models especially for the LHC, NLC and TEVATRON communities, and can provide compatibility of most popular MC models with the new software to be used for current and future HEP experiments.

The RunMC package [2] is a common C++ frond-end of Monte Carlo models which provides a unified approach to generate and analyze very different MC models independent of their native codes. In this approach, the MC output (typically the HEPEVT record) is converted to C++ classes for further analysis or graphical representation (histograms). The graphical user interface (GUI) of this program helps to initialize MC models and histograms, as well as to monitor the event generation.

The RunMC program fully complies with the change in the programming paradigm of data analysis, and meets the requirements of future high-energy experiments. Instead of FORTRAN-based analysis tools, such as PAW [3] and HBOOK [4], it uses the modern CERN C++ analysis packages, CLHEP [5] and ROOT [6].

In this respect, the RunMC program is similar to the JetWeb server [7], which also provides the ability to compare the existing MC models and to confirm the physics assumptions they contain. However, in contrast to JetWeb, the RunMC program was designed as a stand-alone desktop application. Therefore, the user has full access to his calculations and to the program itself.

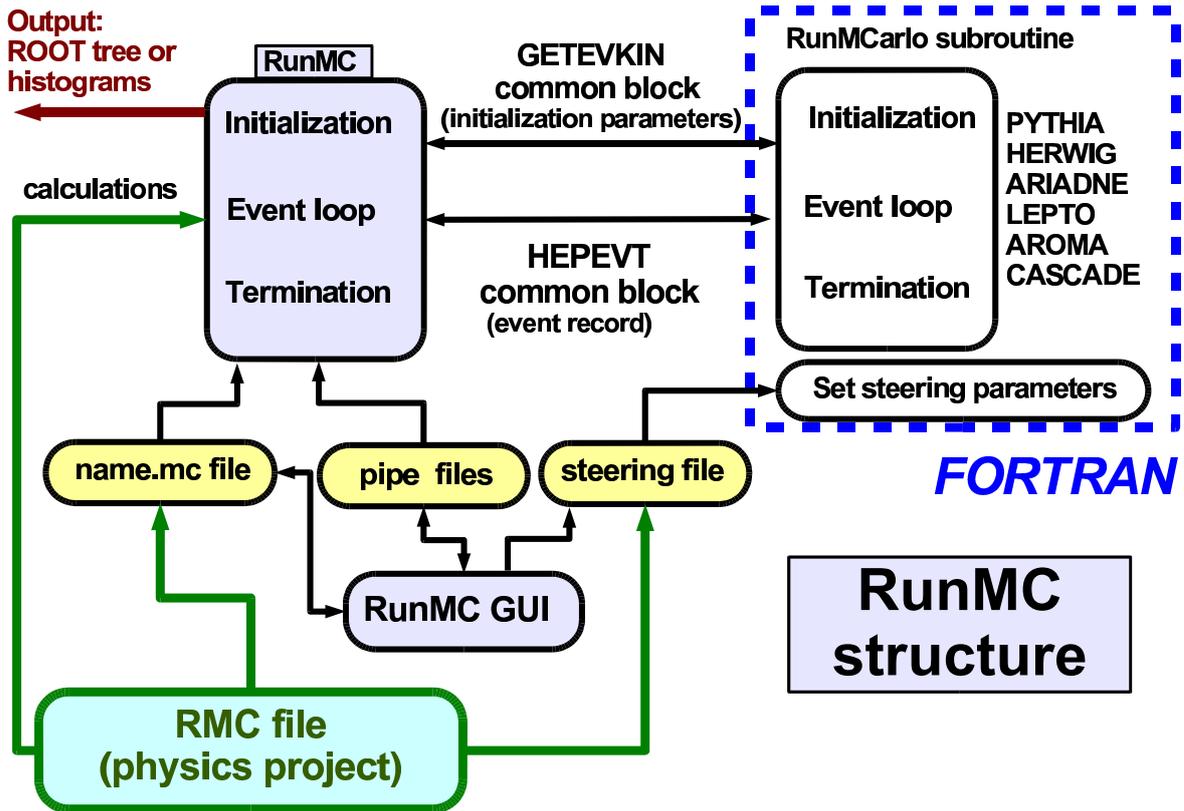


Fig. 1: The diagram shows the structure of the RunMC program.

The RunMC program also provides an interface to the popular HZTOOL package [8], thus many physics calculations from HERA, LEP and TEVATRON can easily be accessed. In addition, within the RunMC approach, the concept of project modules was introduced (in fact, the HZTOOL package is one of such modules). A project file, which can contain external calculations, MC tunings, histogram definitions, etc., can be loaded to RunMC with the same ease as a document can be opened in the Microsoft Word program. The project files are small and platform independent, therefore, it is fairly simple to share complicated physics calculations between scientists in large collaborations.

2 Program structure

RunMC consists of a GUI and several RunMC MC programs. There are two implementations of the RunMC GUI: one is written using the Wide Studio C++ classes [9] and an alternative GUI is based on Java. Due to complete independence of RunMC GUI from RunMC MC programs, one can run jobs in the background without any GUI or pop-up window.

The RunMC programs integrate the C++ ROOT and CLHEP packages with native implementations of MC models. A schematic structure of RunMC is illustrated in Fig. 1. The following MC models are included to RunMC (version 3.3): PYTHIA 6.2 [10], HERWIG 6.5 [11], ARIADNE 4.12 [12], LEPTO 6.5 [13], AROMA 2.2 [14], CASCADE 1.2 [15], PHOJET 1.05 [16], RAPGAP 3.1 [17]. There are several executable RunMC MC programs corresponding to each MC model.

RunMC GUI communicates with the RunMC MC programs using pipe files located in the directory "\$RUNMC/pipes". Here, "\$RUNMC" denotes the installation directory which has to be defined by the user. All the directories to be discussed below are assumed to be located in this directory.

2.1 RunMC GUI

RunMC GUI allows an interaction between the user and RunMC MC programs. At present, two types of RunMC GUI are available: a user interface based on C++ (can be executed with the command “runmc”) and that based on Java (the command “jrunmc”). Below we will describe only the C++ RunMC GUI.

The task of RunMC GUI is to generate the output file “project.mc”, where “project” is a user-defined name of the current calculation. This file contains the most important information for MC running, such as the type of MC model, the number of output events, the type of colliding particles, their energies, RunMC output (histograms or ntuples) etc.

RunMC GUI adopts the following strategy to define the histograms: The window “Variables” contains the names of the variables (with some additional comments) defined for a certain physics project. The user should select the appropriate variable and copy it to the window “Histograms” by clicking on the corresponding variable name. If two one-dimensional histograms are defined, a two-dimensional histogram can be build from these two histograms using RunMC GUI.

The variable names are divided into the three categories: event-based variables (characterizing the event as whole), single-particle densities (filled for each particle/jet; the variable name starts with “@”) and two-particle densities (filled for each particle/jet pair; the name starts with “@@”). The histograms can also be filled in the user-defined subroutine “user-run.cpp”; in this case the naming convention for the variables discussed above is unnecessary.

During the event generation, the ROOT canvas can display the output histograms (up to eight in total) with filled events. The output log from RunMC MC is written to “.analmc.log” (a symbolic link to the “project.log” file). Possible errors are redirected to the file “project.err”, which is constantly monitored by RunMC GUI.

The ROOT histograms are automatically modified at the end of the fill if they are required to be normalized to the total number of events or converted to differential cross sections. Note that there is no need to wait until the end of the current run: once the histogram statistics is sufficient, one can terminate the run by clicking “Stop” on the GUI window. Histograms should be saved in the ROOT file “project.root” for further studies. The style of the histograms can further be modified using the ROOT canvas editor.

2.2 The RunMC MC programs

The RunMC MC programs integrating Monte Carlo models with ROOT C++ classes have the generic names “analmc.MCname”, where “MCname” denotes the MC name. The main C++ function of RunMC MC is located in the file “analmc.cpp” (in the directory “main/src”). The C++ code accesses the HEPEVT common block of a given MC program via a C-like structure. The RunMC MC program receives the initial parameters via the symbolic link “.analmc.ln” pointing to the file “project.mc”.

Each MC model has its own FORTRAN subroutine “runmcarlo” which provides an interface to the native MC code. This interface program (in the file “RUNMC-MCname.f”) is located in the directory “main/mcarlo/MCname”. The task of the subroutine “runmcarlo” is to fill the HEPEVT common block. In addition, some initial settings are done by accessing a C/C++ structure with the initial parameters defined in the “project.mc” file. The main function in “analmc.cpp” calls this interface subroutine and fills the C/C++ structure which represents the complete HEPEVT event record. The output is copied to the class “HEPLIST” which can be accessed by external calculations. The HEPLIST class consists of several vectors based on the LorentzVector vector class (from the CLHEP library) which represents four-momentum of a particle (or a jet). The definition of the HEPLIST class, as well as other include files, can be found in the “main/inc” directory. Note that the user still can access more elementary event records (such as FORTRAN HEPEVT common block) which can be used to transform them to other event classes and physics calculations.

There are several physics packages available inside RunMC MC to transform the original four-momentum vector of particles/jets to the required observable:

- the transformations provided by the physics vector class “LorentzVector” from CLHEP can be used, since a particle or a jet is represented as a general four-vector based on this class;
- the event-shape calculations are available using the package developed by M. Iwasaki [18];
- the longitudinally-invariant k_T algorithm as implemented in C++ [19] can be used for the jet reconstruction. In addition to this package, the JADE and Durham jet algorithms are implemented according to M. Iwasaki [18];
- the Breit frame calculations were implemented for ep deep inelastic scattering.

The physics packages and their documentation are located in the directory “main/physics”.

3 User calculations

For a new physics calculation, the directory “proj” should be modified. This user directory can contain external calculations, steering cards for MC initializations, as well as the standard RunMC functions which are necessary to initialize and fill the histograms.

The user directory should always contain the file “project.mc” created by RunMC GUI. This file can be edited manually without the RunMC GUI program using any text editor. On Linux/Unix, one can load this file to RunMC GUI by executing the command “runmc project.mc” from the shell prompt (or using the option “Projects→read MC” of RunMC GUI).

The directory “proj” can contain steering files “MCname.cards” to redefine initial MC parameters. Such files can be created via RunMC GUI (“MC settings” option). For more flexibility, the MC initialization parameters can also be overwritten by FORTRAN-coded subroutines located in the directory “proj/ini”. If this is not done, the default MC parameters will be selected according to the RunMC option.

To define histograms, user-defined variables should be calculated in the file “proj/user_afill.cpp”. The output of this function is a pointer. The output variable name should always be associated with this pointer. The variable names should be specified in the file “user-name.txt”. It includes the variable names to the list “Variables” accessed by RunMC GUI. Finally, to compile the source codes and to rebuild all RunMC MC programs to take into account changes made in the project source files, one should type “make” in the “proj” directory. All MC programs will be recompiled and RunMC GUI will be updated with new histograms. Then, the command “runmc” (or “jrunmc”) should be executed from the directory “proj” to start RunMC GUI. The main advantage of this approach is that once a necessary variable is defined, new histogram definitions do not require the MC recompilation.

RunMC histograms can also be filled using the conventional method, i.e. in the function located in “user-run.cpp”. In this case, the initialization of histograms is not required, as long as the file “project.mc” defines which histograms should be filled and what presentation style should be used to fill the histograms. The histograms can be initialized in the file “user-init.cpp” using the standard ROOT procedure.

4 Physics calculations as external RMC projects

In order to share complicated analysis calculations or to store them for future use, the directory “proj” can be packed into an external RMC file with the extension “rmc”. For example, “project.rmc” is the RMC file which has the user-defined name “project”. The “proj” directory inside of it has a file “project.mc” with RunMC GUI settings, user-defined external functions, libraries, make files MC steering files, etc.

RunMC GUI can automatically load and recompile such project RMC files (see details in [2]). The user can also save his/her calculations into a RMC file for future analysis. As it was mentioned, the project files are compact and platform independent, therefore, it is fairly simple to share physics calculations between the users, as long as the RunMC package is installed.

At present, several RunMC project files are available on the Web [2] (they are also included in the directory “archive” of RunMC):

- the default project. Only pre-installed variables can be included in the calculations.
- HERA kinematic variables (Q^2 , x , etc.);
- jets at HERA and LHC using the longitudinally-invariant k_T algorithm in the Breit frame. In addition, the ratio of jet cross sections at the parton and hadron levels are calculated (the so-called hadronisation corrections);
- D^* cross sections in ep collisions at HERA;
- cross sections for strange-particle production in ep collisions at HERA;
- the HZTOOL package [8];
- the event-shape variables in e^+e^- at NLC energies;
- several examples of how to visualize tracks and k_T jets in 3D for a single MC event (e^+e^- , ep , pp collisions).

The RMC project files discussed above only illustrate how to set up and to develop new physics calculations in the RunMC framework. For practical applications, these examples should be modified.

5 RunMC ROOT tree analyzer

In addition to the standard functionality of the MC event simulation, RunMC GUI can also use ROOT trees as the input for physics calculations.

The ROOT tree can be generated by selecting the option “HEPEVT” or “RUNMC”, in addition or instead of the ROOT histogram option. Then, the MC events should be generated as usual, but this time a ROOT tree with the extension “.rtup” or “.htup” will be created. Then, RunMC can run over this ROOT tree if, instead of the MC model, the option “RUNMC” or “HEPEVT” is selected. Several ROOT trees can automatically be included in the analysis, as long as they are in the same directory. The analysis of the ROOT trees is very similar to the standard run over MC events. External RMC files can be used to include new calculations, variables and histograms.

The main advantage of the RunMC ROOT tree analyzer is that physics calculations can be validated significantly faster than when RunMC is used to generate events and to fill histograms at the same time. In case of the ROOT trees, RunMC can fill histograms by a factor of ~ 10 – 15 faster, thus the RMC project files can be validated and analyzed more efficiently.

With this additional functionality, the RunMC program can also be used to analyze experimental data if the event record is converted to the appropriate ROOT tree. The data analysis can be performed using exactly the same RMC project files as for the usual MC simulation runs.

References

- [1] M. Bertini, L. Lönnblad and T. Sjöstrand, *Comput. Phys. Commun.* **134**, 365 (2001);
L. Lönnblad, *THEPEG: Toolkit for High Energy Physics Event Generation* (unpublished),
available on <http://www.thep.lu.se/ThePEG/>.
- [2] S. Chekanov, *RUNMC - C++ object-oriented framework for monte carlo models*. Preprint
hep-ph/0411080, *Comm. Phys. Comm.* (in press), available on
<http://hepforge.cedar.ac.uk/runmc/>
<http://www.hep.anl.gov/chakanau/runmc>.
- [3] Application-Software-Group, *PAW: Physics Analysis Workstation*, available on
<http://wwwasd.web.cern.ch/wwwasd/paw/>.
- [4] R. Brun and D. Lienart, CERN-Y250 (1987).
- [5] L. Lönnblad, *Comput. Phys. Commun.* **84**, 307 (1994);
M. Fischler and A. Pfeiffer, *CLHEP - new developments and directions*, in *The proceedings of
International Conference on Computing in High Energy Physics and Nuclear Physics (CHEP
2000), Padova, Italy, 7-11 Feb 2000*. Available on
<http://wwwasd.web.cern.ch/wwwasd/lhc++/clhep/>.
- [6] R. Brun and F. Rademakers, *Nucl. Instrum. Meth.* **A389**, 81 (1997);
R. Brun, F. Rademakers, P. Canal and M. Goto, *ECONF C0303241*, MOJT001 (2003).
- [7] J.M. Butterworth and S. Butterworth, *Comput. Phys. Commun.* **153**, 164 (2003).
- [8] J. Bromley et al., *HZTOOL: a package for MONTE CARLO - data comparisons at heras*, in *Future
Physics and HERA*, eds. Buchmuller, W. and Ingelman, G., Vol. 2, p. 611. Hamburg, Germany,
1996-1997, available on <http://hztool.hep.ucl.ac.uk/>.
- [9] T. Hirabayashi, *The Wide Studio project* (unpublished), available on
<http://www.widestudio.org>.
- [10] T. Sjöstrand, L. Lönnblad and S. Mrenna, *Pythia 6.2: Physics and manual*. Preprint
hep-ph/0108264, 2001.
- [11] G. Corcella et al., *JHEP* **0101**, 10 (2001).
- [12] L. Lönnblad, *Comput. Phys. Commun.* **71**, 15 (1992).
- [13] G. Ingelman, A. Edin and J. Rathsman, *Comput. Phys. Commun.* **101**, 108 (1997).
- [14] G. Ingelman, J. Rathsman and G. Schuler, *Comput. Phys. Commun.* **101**, 135 (1997).
- [15] H. Jung, *Comput. Phys. Commun.* **143**, 100 (2002).
- [16] R. Engel, *PHOJET* (unpublished), 1997, available on
<http://www-ik.fzk.de/~engel/phojet.html>.
- [17] H. Jung, *Comm. Phys. Commun.* **86**, 147 (1995).
- [18] M. Iwasaki, *Jet-EventShape-Finders package* (unpublished).
- [19] J.M. Butterworth, J.P. Couchman, B.E. Cox and B.M. Waugh, *Comput. Phys. Commun.*
153, 85 (2003).

A C++ framework for automatic search and identification of resonances

S. Chekanov

HEP division, Argonne National Laboratory, 9700 S.Cass Avenue, Argonne, IL 60439, USA

E-mail: chekanov@mail.desy.de

Abstract

This paper describes the first proof-of-concept version of a C++ program designed for peak searches in invariant-mass distributions. The program can be useful for searches of new states as well as for the reconstruction of known resonances.

Presently, there is a considerable progress in understanding the rich spectrum of hadron resonances. However, even in case of known baryonic resonances, there are still many open questions. The Particle Data Group (PDG) quotes more than 100 baryonic states, but only half of them are reasonably well established. Recently, the revitalized interest to baryon spectroscopy was triggered by observation of narrow peaks in the $K^\pm n$ and $K_S^0 p$ invariant-mass distributions which can be interpreted as pentaquarks. There is also longstanding interest to other exotic multiquark states, glueballs, hybrids, baryonia, etc. The HERA experiments have active program for searching such states [1]. At the LHC this physics program will continue.

In hadron spectroscopy, searches for new resonances and measurements of known states are based on the reconstruction of invariant-mass distributions of two or more tracks in order to find the rest mass of the originally decaying particle. The usual procedure for such studies is to assign certain masses to tracks, and then to combine their four-momenta to form the invariant-mass distributions. From the observed peaks, one can determine resonance masses, widths and cross sections.

Obviously, a particle identification is highly desirable for any experiment in order to reduce combinatorial background. Without the ability to identify tracks, the combinatorial background rises as $\sim 0.5n^2$, where n is the number of produced particles used in the searches. Thus high-energy experiments have to deal with a significant background for searches in fully inclusive events. For the reconstruction of three- and four-body decays, the combinatorial background is even higher than for the two-body decays.

Particle identification can be achieved using various approaches, such as the energy loss per unit length (dE/dx), silicon-strip detectors, Cherenkov counters etc.. In many cases, the particle identification cannot be perfect. For example, when the dE/dx method is used for pentaquark searches, it is difficult to obtain a high-purity sample of kaons or protons for tracks with large momenta ($p > 1$ GeV) due to significant overlap of the dE/dx bands for different particle species in this momentum range. In this case, several mass assumptions for invariant-mass distributions are needed to be checked in order to exclude peaks due to possible misidentification.

Searches for new resonances using various mass assumptions remain to be a tedious task since no much progress has been made so far to develop a tool which can automate this procedure. For example, the reconstruction of two-, three-, four-body decays involving only three mass assumptions leads to 36 possible non-identical invariant-mass distributions (6 distributions for two-body decays, 10 - for three-particle decays and 20 - for four-particle decays). All these invariant-mass distributions should be reconstructed, analyzed and possible reflections from known PDG states, when different mass assumptions are used, should be disregarded. Obviously, this time consuming work can be simplified and automated.

The program called "SBumps", which is still at the early stage of the development, attempts to accomplish this task. It rather represents the first proof-of-concept version of a program which helps to

perform automatic searches of peaks in invariant-mass distributions. To perform the reconstruction of invariant-mass distributions, the user should specify:

- the event record, i.e. a list of tracks and (optionally) the probabilities that tracks belong to certain particle species. Such probabilities can be obtained using various particle-identification technics;
- the names of particle species used during the mass assignments;
- the statistical significance of expected peaks;
- instrumental invariant-mass resolution;
- how many tracks should be combined to the invariant-mass distributions.

Once the initial conditions are specified, the program runs over the event list and reconstructs all possible invariant-mass distributions with the mass assumptions as specified by the user. SBumps can reconstruct at the same time 2-, 3- and 4- body decays in different combinations. At the end of the run, the program saves all the created invariant-mass distributions to the ROOT histograms [2].

At the second stage, the program analyses the created distributions and attempts to find statistically significant peaks. SBumps uses the build-in ROOT fast peak finder (from the TSpectrum class), which uses a fast deconvolution method [3] based on a Markov approach for peak searching in presence of a background and statistical noise. This algorithm was mainly developed for narrow, high-amplitude peaks which are characteristic for γ -ray physics. Therefore, this algorithm is not completely appropriate for peak searches in the invariant-mass spectra without tunings of the initial parameters of this algorithm. In case of the SBumps package, the deconvolution techniques is only used to identify the so-called seeds, i.e. bins with positions of possible peaks above a smooth background. The number of seeds can be rather large, and not all of them correspond to statistically significant peaks. At this stage, several adjustable parameters are available, such as the resolution of neighboring peaks, the peak sensitivity and the peak thresholds. Such parameters can be set using the steering file before each run.

Next, the program evaluates each seed by calculating the statistical significance, $N(S)/\Delta N$, where $\Delta N = \sqrt{N(B) + N(S)}$, with $N(S)(N(B))$ being the number of the signal (background) events. The calculation was done by comparing the values at the seed positions with the background level, which is determined from the neighboring bins. Then the peaks, which have the statistical significance above the level specified by the user, are considered for further analysis. This part of the program can be further improved, introducing more complicated algorithms. For example, sufficiently broad resonances might be overlooked by the present prescription.

The two-step procedure described above simplifies the peak search since the user normally does not need to deal with tunings of the initial parameters for the fast deconvolution method. At the same time, the method is sufficiently fast and can possibly be extended by introducing other algorithms to identify the peaks. For example, algorithms based on a fitting procedure can also be used; in this case, the seed positions can be used as the initial parameters of the fit functions.

At the third stage, the program attempts to identify the found peaks by comparing them with the masses of known PDG states. For this, a look-up table containing the information on established PDG resonances is used. Since the errors on the reconstructed peaks and the errors on the masses of known resonances taken from the PDG look-up table are known, the program matches the peaks using a simple criteria: $|L|/\Delta L < S$, where S is a free parameter given by the user, L is the distance between the PDG mass and the peak position and ΔL is the error on the reconstructed peak position combined with the error on the mass of known PDG resonances. For this matching procedure, the information on charge of decaying resonance is properly taken into account. In future, the information on specific decay channel can also be taken into account in order to reduce misidentification in the matching procedure.

Figure 1 shows the output of the SBumps program for events generated with PYTHIA Monte Carlo model. This example shows one histogram for two-body decays with two automatically identified peaks corresponding to known states. The required statistical significance for the final peaks was 4σ .

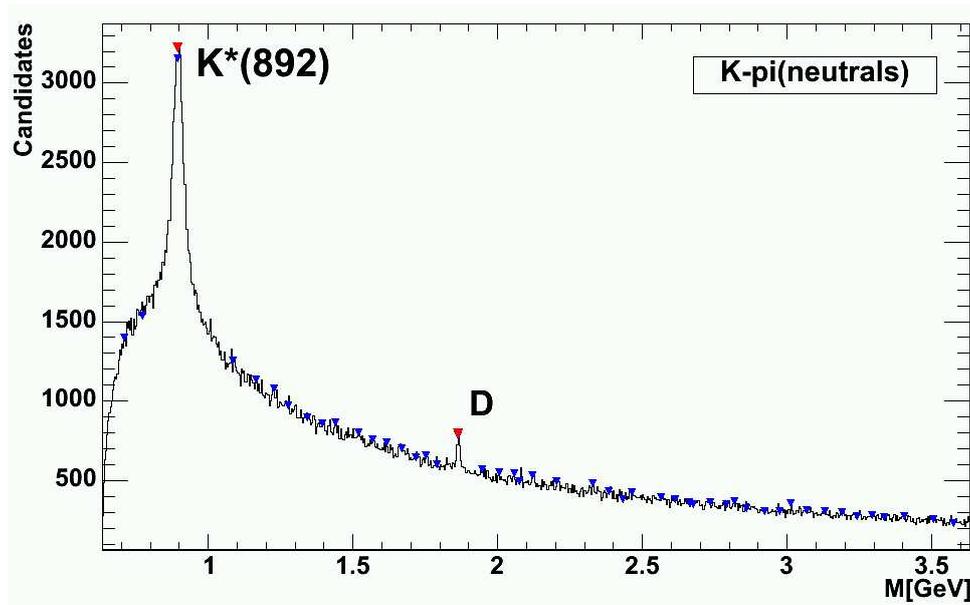


Fig. 1: The two-particle invariant-mass distribution calculated using the events generated with the PYTHIA Monte Carlo model (pp collisions at 1.4 TeV), when two mass hypothesis are used (K^\pm and π mesons). The blue triangle symbols show the seeds used for the calculations of statistical significance. The seeds were found by using the fast deconvolution method. Most of the seed peaks were disregarded after the final calculation of the statistical significance. The program correctly identifies the known PDG states, K^* and D^0 , after comparing the reconstructed peak positions with the PDG look-up table.

This example shows that the program can easily find and identify rather broad resonances (K^*), as well as narrow peaks which do not have high statistical significance.

The proof-of-concept version of the SBumps program is available as a loadable RMC file of the RunMC analysis framework [4]. After loading the “sbumps.rmc” module, RunMC first creates a ROOT tree with Monte Carlo events. The initial conditions are given in the file “sbumps.cards”. The executable file “sbumps.exe” can be used to run over the events. At the end of the calculations, the ROOT browser should display the reconstructed histograms with invariant-mass distributions. The peaks which have the statistical significance above that specified by the user should be labeled.

References

- [1] HERMES Coll., A. Airapetian et al., Phys. Lett. **B 585**, 213 (2004);
ZEUS Coll., S. Chekanov et al., Eur. Phys. J. **C38**, 29 (2004);
ZEUS Coll., S. Chekanov et al., Phys. Lett. **B591**, 7 (2004);
ZEUS Coll., S. Chekanov et al., Phys. Lett. **B610**, 212 (2005);
H1 Coll., A. Aktas et al., Phys. Lett. **B588**, 17 (2004).
- [2] R. Brun and F. Rademakers, Nucl. Instrum. Meth. **A389**, 81 (1997);
R. Brun, F. Rademakers, P. Canal and M. Goto, ECONF **C0303241**, MOJT001 (2003).
- [3] M. Morhac et al., Nucl. Instrum. Meth. **A401**, 113 (1997).
- [4] S. Chekanov, RUNMC - C++ *object-oriented framework for monte carlo models*. Preprint hep-ph/0411080, Comp. Phys. Comm. (in press), available on
<http://hepforge.cedar.ac.uk/runmc/>,
<http://www.hep.anl.gov/chakanau/runmc/>.