A MONTE CARLO PROGRAM FOR LEPTOPRODUCTION

LU TP 80-12

29.10.1980

Gunnar Ingelman and Torbjörn Sjöstrand

Department of Theoretical Physics
University of Lund
Lund
Sweden

Abstract: A FORTRAN program to simulate deep inelastic scattering of muons, or electrons, is described. It is based on a model where perturbative QCD is used to calculate the cross sections to order $\alpha_s$ and the Lund jet model is used for the soft hadronization process.
Contents

1. Introduction 1

2. The model for lepton production 1

3. The program components 2
   3.1. PROGRAM QCDWEIGHTS 2
   3.2. LEPTO-routines 3
       3.2.1. SUBROUTINE LEPBGN 3
       3.2.2. SUBROUTINE LEPPRO(ELFP) 3
       3.2.3. SUBROUTINE LEPGEN(ANU,X) 4
       3.2.4. SUBROUTINE LEPQCD(X,W,QG,QQB,QGMX,QQBMX) 5
       3.2.5. BLOCK DATA 5
       3.2.6. Common blocks 5

4. Structure functions 6

5. Examples of the users program 6

References 8

Appendix 1 : Listing of program QCDWEIGHTS 9

Appendix 2 : Listing of LEPTO routines 15

Appendix 3 : Listing of structure function routines 20
1. Introduction

In this paper we will describe a program that simulates lepton production at high energies. Our basic assumption is that there are two space-time scales involved. On the short scale, characterized by the QCD parameter $1/\Lambda \approx 0.4$ fm, the coloured objects behave as essentially free particles and perturbative QCD is applicable. The long scale corresponds to the soft hadronization where the coloured objects are connected by colour force fields.

Except for the "ordinary" single quark (or antiquark) jets (q-events) there are two dynamically different 2 jet event situations in first order QCD:
(a) the quark (antiquark) radiates off a gluon (qg events)
(b) a gluon is split by the virtual photon into a quark-antiquark pair (qq events)

The cross sections for these different processes are given by Altarelli, Martinelli [1]. For the soft hadronization process we are using the Lund model for quark and gluon jets [2]. The details of how this is implemented in terms of a Monte Carlo program is described in ref. [3]. This model represents a relativistically invariant and causal way of partitioning the energy and momentum of a colour force field among the final state mesons. The dynamics of the force field is that of the massless relativistic string, without any excited transverse degrees of freedom, stretched between a quark and an antiquark. In the case where there is also a gluon present the string is stretched from the quark to the antiquark via a kink which is a representation of the gluon. This means that we do not have three jets fragmenting independently of each other, as in some other models. Hereby we also avoid the problem of joining the jets in the center. In fact, we do not have to select a fixed point in space in any particular Lorentz-frame to connect the jets. This is instead provided in a natural way by the string model we use.

It is important to note that this program conserves energy, momentum (both longitudinal and transverse) and flavour exactly.

2. The model for lepton production

In the following we will often refer to the case of an incoming muon (in the Monte Carlo even assumed to be a $\mu^+$) but the formalism is obviously equally applicable to electroproduction.

This program relies heavily on the use of the programs developed earlier for $e^+e^-$-annihilation [3]. When applying these programs for a lepton production event we note that, for q and qq events, if the virtual photon hits a quark the target remnant is a colour antitriplet. Apart from the particles in the target fragmentation region we expect the same results as if it were an actual antiquark. Thus we
can use the $e^+e^-$ programs by letting the target remnant be represented by an antiquark. For $qq$ events the outgoing quark and antiquark are both connected to the target remnant which is now in a colour octet state, just like a gluon. Again, if we are not studying the target fragmentation region, we can use the same programs with the gluon taking the place of the target remnant.

No electromagnetic radiative corrections are included and the longitudinal structure function is considered to be only a small correction and hence neglected.

The QCD matrix elements for $qg$ and $qq$ events give rise to infrared divergencies for soft and collinear gluons. However, the string model we employ provides a natural cut off for these divergencies. In case of a collinear gluon the energy in the field between the gluon and the quark is too small to break and produce quark-antiquark pairs. Hence, the leading meson will contain energy associated both with the quark and the gluon. For a more detailed description of this and the general features of the model for leptoproduction we refer to [4] where also some results, produced with this Monte Carlo program, are presented.

3. The program components

Below we describe the different elements of the program, which is in reality divided into two separate programs. In the first, the QCD matrix elements are used to calculate the relative probabilities for $qg$ and $qq$ events. These probabilities, or weights, $P_{qg}$ and $P_{qq}$, which are functions of $W$ (the mass of the hadronic system) and the Bjorken $x$-variable are then written on a data file. The second program, the LEPTO-routines, read this data file and simulate complete events.

The advantage of this procedure lies in saving computing time. For a given set of structure functions, $F_2(x,Q^2)$ and $xg(x,Q^2)$, the QCD matrix elements are integrated once and for all.

3.1. PROGRAM QCDWEIGHTS

First order QCD matrix elements [1] are used to calculate the probabilities for $qg$ and $qq$ events in leptoproduction. These probabilities, or weights, are functions of $x$ and $W$ and are therefore calculated on a grid in the $x$-$W$ plane and stored in the arrays $FQG$ and $FQQS$ respectively. (Actually the factor $\alpha/2\pi$ is not included here but added in the LEPGEN routine so that the QCD parameter $\Lambda$ can be easily changed, without having to recalculate the weights).

The weights are obtained by normalizing the integrated matrix elements to the total cross section represented by $F_2(x,Q^2)$. For each $(x,W)$ point the maximum value of the matrix elements in the $z$-$c$ plane is also found and
stored in QGMAX and QQBMAX; z and c represent the two
degrees of freedom in the energy and angle of the gluon.
These maximum values are needed later on in the
simulation process.

All these values are written (unformatted) on logical
file number 20. A listing of the probabilities
(including \( \alpha_s/2\pi \) with \( \Lambda =0.4 \)) is also produced.

Note that the structure functions \( F_2(x,Q^2) \) and \( xg(x,Q^2) \)
must be supplied by the user as FUNCTION subroutines
with the names \( F2(X,Q2) \) and \( XG(X,Q2) \) respectively. The
standard range of \( x \) and \( W \) values covered is \( 0.01 \leq x < 1. \)
and \( 5 < W < 45 \) GeV. If one later on in the simulation
program goes outside this range a warning is given and
an ordinary q jet event is generated. This range can of
course be changed by the user simply by changing the
values given to the arrays \( XX \) and \( WW \). If the dimension
is also changed, then the same changes must, of course,
be made in commonblock QCD in the LEPTO routines. The
integration of the matrix elements is made with an
absolute accuracy determined by the parameter \( \text{ERROR} \)
(\( =0.001 \) by default). For \( Q^2 < Q2\text{MIN} (4. \text{GeV}^2 \) by default)\nthe weights are set to zero. This kind of cut off is
needed since most parametrizations of structure
functions have a lower bound on \( Q^2 \).

3.2. LEPTO-routines

These consist of a set of subroutines that, based on the
already calculated QCD-weights, simulate leptoproduction
events.

3.2.1. SUBROUTINE LEPBGN

A call to LEPBGN is made to initialize the routines,
i.e. to read the QCD-weights from logical file number
20 into the arrays \( POG \) etc. A few values are also set,
e.g. \( \text{PAR(25)} = 0.8 \) gives the width of the Gaussian
primordial \( k_\perp \) distribution \( \langle k_\perp^2 \rangle = \text{PAR(25)}**2 \).

3.2.2. SUBROUTINE LEPPRO(ELEP)

This routine generates a complete event in the lab.
frame for an incoming muon with energy \( \text{ELEP} \) GeV.
Values of \( x \) and \( Q^2 \) are chosen according to the cross
section formula

\[
\frac{d\sigma}{dx dQ^2} = \text{const} \left( 1 - y + y^2/2 \right) \frac{F_2(x,Q^2)}{x Q^2}
\]

where

\[
y = \frac{v}{E_\mu} = \frac{Q^2}{2 ME_\mu x}
\]

and assuming the following cuts:
(a) $Q^2 > \text{CUT}(1)$ 
(b) $\text{CUT}(2) < x < \text{CUT}(3)$
(c) $\text{CUT}(4) < \nu < \text{CUT}(5)$
(d) $W^2 > \text{CUT}(6)$
(e) $\theta_{\mu} > \text{CUT}(7)$
(f) $E_{\mu} > 1 \text{ GeV}$

(4 GeV$^2$ by default)
(0 and 1 by default)
(1 and 1000 by default)
(5. GeV$^2$ by default)
(0. degrees by default)
(energy of scattered lepton)

These cuts can easily be changed via the common block CUTS.

By a call to LEPGEN the hadron shower is generated and then a rotation of the jet axis is made from the positive z direction to the angles $(\theta, \phi)$ of the virtual photon. Finally, the scattered muon is added as the last (N:th) particle in the arrays K and P in which information on produced particles is stored. Note that all particles from the fragmentation process are kept. To get rid of e.g. unstable and/or neutral particles one can use CALL EDIT(...), see page 40 in reference [3].

The coordinate system employed is right-handed with the incoming muon along the positive z direction. The azimuthal angle of the virtual photon is chosen at random. The azimuthal angle of the hadron event plane around the direction of the virtual photon is also randomly chosen. (Note that this is not entirely correct [5], the modifications expected should however be minor for most practical purposes.)

3.2.3. SUBROUTINE LEPGEN(ANU,X)

Given values of the arguments $\nu$ and $x$ the hadronic jet(s) is generated. The type of event $q$, $qg$ or $q\bar{q}$ is determined from the QCD-weights. For events with a gluon present the energy and angle of it is determined by the QCD matrix elements (at this point the values QGMAX and QQBMAX are needed).

For $q$ and $qg$ events the flavour of the struck quark is determined from the different quark structure functions. This means that not only $F_2(x,Q^2)$ and $xg(x,Q^2)$ have to be supplied by the user, but also $xu(x,Q^2), xd(x,Q^2), x\bar{u}(x,Q^2), xs(x,Q^2)$ and $xc(x,Q^2)$ have to be given as FUNCTION subroutines. Note that $xu$, $xd$ are the valence distributions and $x\bar{u}$, $xs$, $xc$ the sea distributions for each of the quarks $u, \bar{u}, d, \bar{d}, s, \bar{s}, c, \bar{c}$ separately and not their sum (u-d symmetric sea assumed).

The event is generated in the quark-antiquark (or quark-antiquark-gluon) center of mass frame. A primordial $k_\perp$ is chosen according to a Gaussian distribution ($\langle k_\perp^2 \rangle = \text{PAR}(25)^*2$) and given to the event by swinging its axis (in the cm frame) correspondingly. Finally a boost along the z axis is made to transform to the lab frame (the rotation to the angle of the virtual photon is made after the
return to LEPPRO).

3.2.4. **SUBROUTINE LEPOCD**(*X*,*W*,*QG*,*QQB*,*QGMX*,*QQBHX*)

To get the weights for *qg* and *qq* events for a given value of (*x*,*W*) this routine makes a linear interpolation of the values on the *x*-*W* grid.

3.2.5. **BLOCK DATA**

Sets default values for the cuts.

3.2.6. **Commonblocks**

**COMMON /JET/ N,K(250,2),P(250,5),NC,KC(10),PC(10,4)**

Stores the relevant data about a generated event.

(a) *N* is the number of particles generated and stored in the *N* first rows of the *K* and *P* arrays.

(b) *K(1,1)* describes the origin of the *I*:th particle. *K(1,2)* gives the particle type according to table 1 in [3].

(c) *P(1,1), P(1,2), P(1,3)* give the particle momentum *p_x, p_y, p_z* in GeV/c. *P(1,4)* and *P(1,5)* give the particle energy and mass respectively.

(d) *NC* is the number of primary partons (quarks and gluons) generated.

(e) *KC(IC)* gives the flavour of the *IC*:th parton.

(f) *PC(IC,1),P(IC,2),P(IC,3)* give the parton momentum and *P(IC,4)* the parton energy.

**COMMON /DATA1/ IFR,PUD,PSI,SIGMA,QMAS(7),PMAS(96),PAR(25)**

Stores the most frequently used parameters and data.

(a) *IFR* (by default 2) is used to choose fragmentation scheme, for 1 it gives an "extended Field-Feynman" and for 2 it gives the Lund model.

(b) *PUD* (by default 0.4) is the fraction of *u*-pairs created in the field during the jet fragmentation and is assumed to be the same for *d*-pairs, with the rest being *s*-pairs.

(c) *PSI* (by default 0.5) is the fraction of primary mesons with spin 1.

(d) *SIGMA* (by default 0.44) is the standard deviation of the transverse momentum (*p_x* and *p_y* components) of primary mesons formed in the field.

(e) *PAR(6)*, *PAR(7)* (by default 4., 0.4 GeV) give the number of flavours and the QCD parameter *Λ* to be used in the formula for *q_s*.

(f) *PAR(24)* is a flag for the type of the event generated, 1. for *q* jet, 2. for *qg* jet and 3. for *qq* jet.

(g) *PAR(25)* (by default 0.8) gives the width of the Gaussian primordial *k_⊥* distribution.
\[ \langle k_\perp^2 \rangle = \text{PAR}(25) \times 2 \].

For a more detailed description of these two common blocks we refer to ref. [3].

COMMON /CUTS/ F2MAX, CUT(7), X, Y, ANU, W2, Q2

Its purpose is to make it easy to apply cuts on the events and also make some basic kinematical variables available. For details of the cuts see section 3.2.2. X, Y are the usual scaling variables, ANU = \nu, W2 = \nu^2 and Q2 = Q^2.

4. Structure functions

A set of structure functions has to be supplied to the programs. The user can have his own favourite set or use one of the two supplied standard sets. These are the Q^2-dependent parametrizations made by Buras and Gaemers [6] and Glück, Hoffmann and Reya [7].

The structure functions should be in the form of FUNCTION subroutines with the following names:
(a) F2(X, Q2)
(b) XU(X, Q2) = xu valence distribution
(c) XD(X, Q2) = xd valence distribution
(d) XUB(X, Q2) = xu = \bar{x}u = xd = \bar{x}d, i.e. u, d symmetric sea
(e) XS(X, Q2) = xs = \bar{x}s strange sea distribution
(f) XC(X, Q2) = xc = \bar{x}c charmed sea distribution
(g) XG(X, Q2) = xg gluon distribution

5. Examples of the user's program

In this first example complete events are generated by the LEPTO routines:

COMMON /JET/ N, K(250, 2), P(250, 5), NC, KC(10), PC(10, 4)
COMMON /CUTS/ F2MAX, CUT(7), X, Y, ANU, W2, Q2

CALL LEPBGN
DO 100 I = 1, NEVENT
CALL LEPPRO(280.)
CALL EDIT(3, 0, 0., 0.) ; keep only charged,stable particles
; analysis of event
100 CONTINUE
STOP
END

In this second example the kinematical variables are supplied by the user, e.g. from data, and hence LEPPRO is by-passed:

COMMON /JET/ N, K(250, 2), P(250, 5), NC, KC(10), PC(10, 4)
CALL LEPBGN
DO 100 I=1,NEVENT
  
  ANU = ...
  X = ... 
  
  CALL LEPGEN(ANU,X)
  CALL EDIT(3,0,0.,0.) ; keep only charged,
  
  ; stable particles
  
  THETA = ...
  PHI = ...
  
  CALL ROTBST(THETA,PHI,0.,0.,0.) ; rotate hadron
  
  ; shower to photon axis
  
  ; analysis of event
  
100 CONTINUE
STOP
END

The program should be loaded together with the LEPTO routines, the JET routines of ref. [3] and the desired set of structure functions and the corresponding data file with QCD-weights specified as logical unit 20. As an example of needed computing time we mention that on an IBM 370/168 it takes about 30 CPU-seconds to generate 1000 events.
References

1. G. Altarelli, G. Martinelli
   Phys. Lett. 76B (1978) 89

2. B. Andersson, G. Gustafson, C. Peterson
   Z. Physik C1 (1979) 105
   B. Andersson, G. Gustafson
   Z. Physic C3 (1980) 22
   B. Andersson, G. Gustafson, T. Sjöstrand
   LU TP 80-1 (to be published in Z. Physik C)

3. T. Sjöstrand: A Monte Carlo program for quark and
   gluon jet generation, LU TP 80-3

4. B. Andersson, G. Gustafson, G. Ingelman, T. Sjöstrand:
   On high energy leptoproduction, LU TP 80-6

5. H. Georgi, H.D. Politzer

6. A.J. Buras, K.J.F. Gaemers

7. M. Glück, E. Hoffmann, E. Reya:
   Scaling violations and the gluon distribution of
   the nucleon, Dortmund preprint DO-TH 80/13
A Monte Carlo program for leptoproduction

Appendix 1: Listing of program QCDWEIGHTS

C
C PROGRAM QCD-WEIGHTS
C
C USES FIRST ORDER QCD MATRIX ELEMENTS TO CALCULATE THE PROBABILITY
C FOR QUARK-GLUON AND QUARK-ANTIGLUON EVENTS IN LEPTOPRODUCTION.
C
C REQUESTED FUNCTION'S : F2(X,Q2) AND XG(X,Q2)
C
C C
C*****************************************************************************
C
C EXTERNAL QBAR,QGLHOS,FL,PU
C DIMENSION XX(20),WW(17),PQG(20,17),PQG(20,17),
C & QGHAX(20,17),QGHMAX(20,17)
C
C...GIVE X AND W VALUES TO ARRAYS XX AND WW WITH DIMENSION NXX
C...AND NWW RESPECTIVELY.
C DATA XX/0.025,0.05,0.075,1.,1.25,1.5,1.75,2,2.25,3,3.5,4,4.5,6.
C & 8.5,6.7,8.8,9.99/
C DATA WW/5.7,5.10,12.5,15,17.5,20,22.5,25,27.5,30,32.5,35,37,5,40,42.5,45,60/
C DATA NXX,NWW/20,17/
C
C...SET REQUIRED ABSOLUTE ACCURACY OF INTEGRATION, MINIMUM Q**2
C...AND QCD-LAMDA.
C DATA ERROR,Q2MIN,QLAMDA/0.001,4.0,0.4/
C
C...CALCULATE VALUES FOR CUTS ON SOFT AND COLLINER GLUONS
C CH(1)=QMAS(1)**2+10.*QMAS(1)*PAR9+12.5*PAR9**2
C CH(2)=8.*PAR9**2
C P2MAX=0.
C WRITE(6,200)
C DO 100 IWW=1,NWW
C WW=WW+W
C WW=WW+W
C DO 100 IWW=1,NWW
C WW=WW+W
C ANU=(ANU-8)/((2.*R38**2+1.)/)
C Q2=2.*R38*ANU**2
C Q=(QRT(Q2)
C
C...CALCULATE ALPHA-STRONG/2/PI
C ALPS2=3.((2.*ALPHA(Q/QLAMDA))
C ALFA=ALPS2*3.14159
C QBAR=0.
C QHAX=0.
C QHUX=0.
C IF(Q2,L=QHUX) GOTO 50
C P2Q2=P2(X,Q2)
C TOTAL=P2Q2/X
C C...FIND MAXIMUM OF F2
C IF(P2Q2,GT,F2MAX) F2MAX=P2Q2
C C...INTEGRATE MATRIX ELEMENT
C EPS=ERROR
C CALL GADAP2(X,1.,FL,PU,QBAR,EPS,QBAR)
C CALL GADAP2(X,1.,FL,PU,QBAR,EPS,QBAR)
C EPS=ERROR
C
C*****************************************************************************
C
C CALL GADAP2(X,1.,FL,PU,QGLHOS,EPS,QU)
C 50 PQ QB=AQB/ALSP
C QBHAX(X,1,4)=QHAX
C PQG(1,1,4)=QG/ALSP
C QGHAX(1,1,4)=QGH
C PQI=1.-QG-QH
C WRITE(6,300) W,X,Q2,ANU,ALFA,Q,Q,PQG,QGHAX,QUH
C 100 CONTINUE
C WRITE(6,400) X,X,W,ERROR,Q2MIN,QLAMDA,F2MAX

C...WRITE RESULTS (UNFORMATTED) ON LOGICAL FILE NUMBER 20
C WRITE(20) XNN,WW,XX,WW,WW,PQG,QHAX,QUHAX,F2HAX
C 200 FORMAT(X1,,//,10X,"X","W","W",WW,PQG,QHAX,QUHAX,F2HAX)
C 300 FORMAT(X2,F10.1,F10.1,F10.1,F10.4,2E12.4)
C 400 FORMAT(X1,,//,17,"X","W","W",WW,PQG,QHAX,QUHAX,F2HAX),
C & /*,T5,T5,T20,FS4,6,FS8,2,//
C STOP END
C
C REAL FUNCTION FL(Y)
C C...GIVE LOWER BOUNDARY FOR INTEGRATION IN C=COS(THETA)
C FL=1.
C RETURN END
C
C REAL FUNCTION F(Y)
C C...GIVE UPPER BOUNDARY FOR INTEGRATION IN C...F=1.
C RETURN END
C
C REAL FUNCTION QBAR(Z,C)
C C
C C CALCULATE VALUE OF QCD MATRIX ELEMENT AS A FUNCTION OF Z AND C
C FOR QUARK-ANTIGLUON EVENTS, I.E. PHOTON-GLUON FUSION PROCESS.
C C ALSO MAKE CUTS AGAINST COLLINER GLUONS.
C C
C*****************************************************************************
C
C COMMON/VAR/X,ANU,Q2,W,ALSP,TOTAL,QGUX,QUHUX,QMAS(7),CM,PAR9
C U*X/Z
C EM3=Q2*(1.-1./Z)
C EM1=Q2*(Z-X)*(1.-C)/(X*(1.+2.*Z-1.)/)
C EM2=Q2*(Z-X)*(1.+C)/(X*(1.+2.*Z-1.)/)
C IF(EM1.LE.CM(1),OR.EM2.LE.CM(1)) GOTO 100
C PMASS=U*2.*R38*ANU+(1.*R38**2)*-Z2
C S0=0.
C DO 20 IFL=1,4
C C...CHECK THAT INVARIANT MASS OF PHOTON+GLUON IS BIG ENOUGH TO
C C...PRODUCE QUARK-ANTIGLUON PAIR
C IF(PMASS-2.*CM(1)/PMASS)**2 GOTO 20
C CHARGE=1.
C IF(FLF,0.,1.) CHARGE=-1.
C IF(FLF,0.,1.) CHARGE=-1.
C IF(FLF,0.,1.) CHARGE=-1.
A Monte Carlo program for leptoproduction

C MAKE EXTRA CUTS FOR THE HEAVIY CHARMED QUARK BY USING THE C THRESHOLD (1-UTH**2/2**2) ; UTH**2 = 22. GEV**2 C IF (X2 .LT. 22.) GOTO 20 C XCH=1.5.*PAR9**5.*PAR9+4.*QMAS (IFL)) / (2.*U2) C IF (X2 .GT. XCH .OR. X1 .GT. XCH) GOTO 20 C CHARGE = 4.* (1.-22./U2) C 10 EQ = SQ + CHARGE C CONTINUE C P=SQ**9./Z**2**2*(U1+2.*Z-1.)*C)**2**2 C (Z**2**1.)/(1.-C)**2.*Z**2**1.)/(1.+C)**2**2 C 4.* (Z**2**1.)/(1.+Z)**2**2 C QBAR=P*ALS2P/TOTAL C IF (P.GT.QBARH) QBAR = P C RETURN C 100 QBAR=0. C RETURN C REAL FUNCTION QGLUON(Z,C) C ******************************************** C C CALCULATE VALUE OF QCD MATRIX ELEMENT AS A FUNCTION OF Z AND C C FOR QUARK-GLUON EVENTS, I.E. RADIATION OF A GLUON. C ALSO MAKE CUTS AGAINST SOFT AND COLLILINEAR GLUONS. C ******************************************** C COMMON/VAR/X,ANU,Q2,W2,ALS2P,TOTAL,QGMX,QQBMX,QMAS (7),CM (2),PAR9 U=Z/E Z/EM2=Q2**(1.-Z)/Z IF (ENZ.LT.CH (1)) GOTO 100 EM1=Q2**(1.-Z)/Z*(1.-C)/(X**2**1.)/(1.+C) EM2=Q2**(1.-Z)/Z*(1.-C)/(X**2**1.)/(1.+C) IF (ENZ.LT.CH (1)) EM2=EM2*EM2*EN1.LT.CH (2)) GOTO 100 C P=FZ (Q2)/Z**2**2/(1.)*Z**2**1.)/(1.-Z)**2**2 C (1.+C)**2**2/(1.+Z)**2**2 C QQBMX=P*ALS2P/TOTAL C IF (P.GT.QBMX) QQBMX = P C RETURN C 100 QQBMX=0. C RETURN C BLOCK DATA COMMON/VAR/X,ANU,Q2,W2,ALS2P,TOTAL,QGMX,QQBMX,QMAS (7),CM (2),PAR9 C GIVE QUARK MASSES. PAR9=PAR (9) IN REF. 3. C DATA QMAS/5.3,5.1,6.5,20.,275./PAR9/A/ C A Monte Carlo program for leptoproduction


SUBROUTINE GADAP2(AO,BO,FL,FR,FU,F,EPS,SUM) COMMON/GADAP1/ NUM,IFU EXTERNAL F,FL,FR,FU DIMENSION A (300),X (300),FL (300),F2 (300),F3 (300),S (300),N (300) I FORMAT(16H GADAP: I TOO BIG) DSUM(F1,F2,F3,A,B,A,B)=S.1R.1S*(ABAB)*F1+1.6*F2+F3) IP(EPS,L.T.1.4E-6) EPS=1.4E-6 RED=1.4 L=1 I=1 SUM=0. C=SQR (15.)/5. A (1)=AO B (1)=BO X=0.5*(1+C)A0+0.5*(1-C)BO AY=FL (X) BY=FU (X) F2 (1)=FU (X) X=0.5*(A+BO) AY=FL (X) BY=FU (X) F2 (1)=FU (X) X=0.5*(A+C)A0+0.5*(1+C)BO AY=FL (X) BY=FU (X) F3 (1)=FU (X) EPS=3 I=I DSUM(F1,F2,F3,A,B) EPS=3 A (1)=BO B (1)=AO CONTINUE L=I+1 N (L)=3 EPS=EPS RED A (I+1)=A (I)*C(R-I)-A (I) B (I+1)=B (I) A (I+2)=A (I)+5*I-A (I+1) B (I+2)=A (I+1) A (I+3)=A (I) B (I+3)=A (I+2) W1=A (I)+W (I)-A (I))/5. UZ=2.*W1-A (I)+A (I+2))/5.
A Monte Carlo program for leptoproduction

X=A(1)+B(1)-W1
AY=FL(X)
BY=FU(X)
F1(I)=FGADAP(X,AY,BY,F,EPS)
F2(I)=F3(I)
X=A(I)-A(I+2)+W1
AY=FL(X)
BY=FU(X)
F3(I)=FGADAP(X,AY,BY,F,EPS)
X=W2
AY=FL(X)
BY=FU(X)
F1(I)=FGADAP(X,AY,BY,F,EPS)
F2(I)=F2(I)
X=A(I)+A(I+2)-W2
AY=FL(X)
BY=FU(X)
F3(I)=FGADAP(X,AY,BY,F,EPS)
X=A(I)+A(I+2)-W1
AY=FL(X)
BY=FU(X)
F1(I)=FGADAP(X,AY,BY,F,EPS)
F2(I)=F1(I)
X=W1
AY=FL(X)
BY=FU(X)
F3(I)=FGADAP(X,AY,BY,F,EPS)
IF(IFU=FU+6)
IF(IFU GT 5000) GOTO 130
S(I)=SUM(F1(I),F2(I),F3(I),A(I+1),B(I+1))
S(I+2)=SUM(F1(I+2),F2(I+2),F3(I+2),A(I+2),B(I+2))
S(I+3)=SUM(F1(I+3),F2(I+3),F3(I+3),A(I+3),B(I+3))
SS=S(I)+S(I+2)+S(I+3)
I=I+3
IF(IFU GT 5000) GOTO 130
SOL=S(I-3)
IF ((ABS(SOL-SS).GT.EPS*(1.+.ABS(SS))/2.) GOTO 100
SUM=SUM+SS
1=1-4
N(L)=0
L=L-1
110 CONTINUE
IF(L EQ 1) GOTO 130
N(L)=N(L)-1
EPS=EPS/RED
IF (N(L).NE.0) GOTO 100
1=1
L=L-1
GOTO 110
120 WRITE (6,1)
130 RETURN

FUNCTION FGADAP(X,AO,BO,F,EPS)
COMMON/GADAP1/ NUM, IFU
EXTERNAL F
DIMENSION A(300), B(300), F1(300), F2(300), F3(300), S(300), N(300)
1 FORMAT (15H GADAP: I TOO BIG)
SUM(F1,F2,F3,AO,BO)=S(0)*10.*ABS(AO)*F1+1.6*F2+F3
IF (EPS LT 1.0E-8) EPS = 1.0E-8

END

RED=1.4
L=1
I=1
SUH=0.
C=SOR(T(15.)/5.
A(I)=A
B(I)=B0
F1(I)=F(X, 0.5*(1+C)*A0+0.5*(1-C)*30)
F2(I)=F(X, 0.5*(A0+30))
F3(I)=F(X, 0.5*(1-C)*A0+0.5*(1+C)*30)
IFU=3
S(I)=SUM(F1(I),F2(I),F3(I),A0,B0)
100 CONTINUE
LW=L+1
N(L)=3
EPS=EPS*RED
A(I)=A(I)+C*(B(I)-A(I))
B(I+1)=B(I)
B(I+2)=A(I+1)
A(I+3)=A(I)
B(I+3)=A(I+2)
W(I)=A(I)+(B(I)-A(I))/5.
U2=2.*U1-(A(I)+A(I+2))/2.
F1(I)=F(X,A(I)+B(I)-U1)
F2(I)=F(X,A(I)+B(I)-U2)
F3(I)=F(X,A(I)+B(I)-U3)
F1(I+2)=F(X,U2)
F2(I+2)=F(X,U2)
F3(I+2)=F(X,U2)
F1(I+3)=F(X,U3)
F2(I+3)=F(X,U3)
F3(I+3)=F(X,U3)
IFU=IFU+6
IF(IFU GT 5000) GOTO 130
S(I)=SUM(F1(I+1),F2(I+1),F3(I+1),A(I+1),B(I+1))
S(I+2)=SUM(F1(I+2),F2(I+2),F3(I+2),A(I+2),B(I+2))
S(I+3)=SUM(F1(I+3),F2(I+3),F3(I+3),A(I+3),B(I+3))
SS=S(I)+S(I+2)+S(I+3)
I=I+3
IF(IFU GT 5000) GOTO 130
SOL=S(I-3)
IF ((ABS(SOL-SS).GT.EPS*(1.+.ABS(SS))/2.) GOTO 100
SUM=SUM+SS
1=1-4
N(L)=0
L=L-1
110 CONTINUE
IF(L EQ 1) GOTO 130
N(L)=N(L)-1
EPS=EPS/RED
IF (N(L).NE.0) GOTO 100
I=1
L=L-1
GOTO 110
120 WRITE (6,1)
130 FGADAP=SUM
EPS=EPS/RED
RETURN
END
SUBROUTINE LEPBGN
C
INITIALIZE LEPTO-ROUTINES. READ QCD-WEIGHTS FROM LOGICAL FILE 20.
PAR(25) GIVES WIDTH OF PRIMORDIAL HT-DISTRIBUTION.

C
COMMON /CUTS/ F2MAX,CUT(7),X,Y,ANU,W2,Q2
COMMON /QCD/ NXX,NWW,XX(20),WU(17),PPG(20,17),PQOB(20,17),
& QMAX(20,17),QBMAX(20,17),CM(2)
COMMON /DATA1/ IST,IFR,PUD,PSI,SIGMA,QNAS(7),QNAS(96),PAR(25)
READ(20) NXX,NWW,XX,WU,PPG,PQOB,QBMAX,QBMAX,F2MAX
PAR(6)=4.
PAR(7)=4.
PAR(9)=8.
PAR(25)=8.
CM(1)=QHAS(1)**2+10.*QHAS(1)*PAR(9)+12.5*PAR(9)**2
CM(2)=B.*PAR(9)**2
RETURN
END

SUBROUTINE LEPPPRO(ELEP)
C
CHOOSE X,Y AND Q2 FROM CROSS SECTION FORMULA. MAKE CUTS ON
KINEMATICAL VARIABLES. GENERATE PARTICLES VIA CALL TO LEPGEN.
C
COMMON /JET/ N,K(250,2),P(250,5),NC,KC(10),PC(10,4)
COMMON /DATA1/ IST,IFR,PUD,PSI,SIGMA,QHAS(7),QHAS(96),PAR(25)
COMMON /CUTS/ F2MAX,CUT(7),X,Y,ANU,W2,Q2
COMMON /CUTS/ F2MAX,CUT(7),X,Y,ANU,W2,Q2
Q2=CT(1)/RUNF(0)
XMIN=CUT(1)/(2.*Q38*ELEP)
110 X=XMIN**RUNF(0)
IF(XL,CUT(2),OR.XL,CUT(3)) GOTO 110
X=1.-X
IF(P2(W2,Q2),LT,F2MAX,RUNF(0)) GOTO 100
Y=Q2/(2.*Q38*ELEP)*X
IF(1.-Y+5**2*RUNF(0)) GOTO 100
ELEP**2=(1.-Y)
IF(ELEP**2).LE.1. GOTO 100
ANU=ELEP**Y
IF(ANU,LT,CUT(4),OR.ANU,LT,CUT(5)) GOTO 100
Q=SQRT(Q2)
W=2.*Q38*ANU*X + Q2
IF(WL,LW(6)) GOTO 100
W=SLR(W2)
THET=ACOS(1.-Q2/(2.*ELEP**ELEP))
IF(THET*180./3.1415927,L.CUT(7)) GOTO 100
CALL LEPGEN(ANU,X)
C
A Monte Carlo program for lepto-production

IF(RF2.GT.XC8) GOTO 40

C...APPLY THRESHOLD FACTOR FOR CHARMED QUARKS
IF((1.-22./W2).LT.RANF(0)) GOTO 30
IF(L=4) SIGNS
GOTO 70
40 IF(RF2.GT.XC8*X52) GOTO 50
IF(L=3) SIGNS
GOTO 70
50 IF(RF2.GT.XC8+XS2+XUB10) GOTO 60
IF(L=MAX(1,LNT(5.*RANF(0)-2.),3)) SIGNS
GOTO 70
60 IF(RF2.GT.XC8+XS2+XUB10+XD1) GOTO 70
IF(L=2)
70 IF(LAABB(SLFL)) GOTO 200
100 JETTYPE=1

C...GENERATE QUARK-JET EVENT
CALL QD(JETFLN,V)

GOTO 200
200 JETTYPE=2

C...GENERATE QUARK-GLUON JET EVENT, CHOOSE Z AND C ACCORDING TO QCD

C...MATRIX ELEMENTS AND APPLY CUTS FOR SOFT AND COLLINER GLUONS
210 Z=(1.-X)\*RANF(0)
EM2=Q2/(1.-Z)/Z
IF(EM2.LT.CM(1)) GOTO 210
C=2.*RANF(0)-1.
EM1=Q2/(1.-Z)*((1.-Z)/(1.-C)/(1.-Z)/(1.+2.*Z-1.0)C))
EM2=Q2/(1.-Z)*(1.+C)/(1.+2.*Z-1.0)C)
EM2=EM2/EM2(1,3)
GOTO 210
SIG=FZ[X/F2(2,1),2,0]*Z
AZ=(Z-A)2/(1.+C2)\*\(1.-Z\*2)\*(1.+C)\*(1.+C)/(1.+2.*Z-1.0)C)
IF(SIG.LT.QGMX/RANF(0)) GOTO 210
X1=-EM1-QMAS(IFLA)*2/N2
X2=-EM2-QMAS(IFLA)*2/N2
X3=-X1+X2
IF((X1.1)2\*(X1.1-X2.1)2\*(X1.1-X2.1)2\*(X1.1-X2.1)2) GOTO 100
XCU1=1.-Z\*PAR(9)*(5.\*PAR(9)+4.*QMAS(IFLA))/2.\*U2
IF(X2.GT.XCUT OR X1.GT.XCUT) GOTO 100
CALL QQGQ(IFL,W,X1,X2)
CALL ROTBST(ACOS(-PC(2,3)/SQR(CT(PC(2,3)**2+PC(2,1)**2)),6,2832/RANF(0),0.0,0.0)
GOTO 500
300 JETTYPE=3

C...GENERATE QUARK-ANTIQUARK JET EVENT, CHOOSE Z AND X ACCORDING TO QCD

C...QCD MATRIX ELEMENTS AND APPLY CUTS ON COLLINER GLUONS
310 Z=(1.-X)\*RANF(0)
EM2=Q2/(1.-Z)/Z
EM1=Q2/(1.-Z)*((1.-Z)/(1.-C)/(1.+2.*Z-1.0)C))
EM2=Q2/(1.-Z)*((1.+C)/(1.+2.*Z-1.0)C)
EM1=EM1/EM2/EM2(1,3)
GOTO 310
U=X/Z
QMAS=U**2\*938\*ANU+(U**2\*938)**2\*Q2
SO=0.
DO 350 IFLA=1,4
IF(PQUAMS.LT.2.**QMAS(IFLA)**2) GOTO 350
CHARGE = 1.
IF(IFLAM=0.1) CHARGE = -1.
IF(SO=0) IFLA=1,4
GOTO 350
350 Continued

C...THRESHOLD FACTOR FOR CHARMED QUARK
IF(V2.LT.22.) GOTO 350
XCU1=1.-Z\*PAR(9)*(5.\*PAR(9)+4.*QMAS(IFLA))/2.\*U2
IF(X2.GT.XCUT OR X1.GT.XCUT) GOTO 350
CHARGE = 4.*(1.-22./U2)
340 SQ = SQ + CHARGE
350 CONTINUE
SIG=SQ/9.*XG(U/2)/U/(1.+2.\*Z-1.0)C**2
Z=(Z-A)2/(1.+C2)\*(1.+C)/(1.+2.\*Z-1.0)C)
Z=(Z-A)2\*(1.+C)/(1.+2.\*Z-1.0)C)
IF(RANF(1.-Z)/2.**Z2) GOTO 310
IF(SIG.LT.QGBM/RANF(0)) GOTO 310
SIG=SQ/RANF(0)

C...CHOOSE FLAVOUR
IFL=FL(4.) IFLA=2
IF(IFL=FL(5.) IFLA=3
IF(IFL=FL(6.) IFLA=4
IFL=IFL/RANF(1.
X1=-EM1-QMAS(IFLA)*2/N2
X2=-EM2-QMAS(IFLA)*2/N2
X3=-X1+X2
CALL QQGQ(IFL,W,X1,X2)
CALL ROTBST(ACOS(-PC(2,3)/SQR(CT(PC(2,3)**2+PC(2,1)**2)),6,2832/RANF(0),0.0,0.0)
GOTO 500
500 Continued
C...TAU/BETA/PRIMORDIAL KT
CALL PT(LOCAL*PX/2+FY**2)
PHI=SIG(ACOS(PX/IFL),FY)
CALL ROTBST(ATHAPK/AES(CT(JETYPE,3)),PHI,0.0,0.0)
C...MAKE BOOST ALONG Z-AXIS TO LAB-SYSTEM
CALL ROTBST(0.0,0.0,0.0,1./SQR(1.+2.**Q1))
PAR(24)=JETYPE
RETURN
END

SUBROUTINE LEPGOC(X,W,QG,QQB,QM2,QQBM)
C ************* C
C TO GET THE QCD-WEIGHTS FOR GIVEN VALUES OF (X,W) MAKE A LINEAR
C INTERPOLATION OF THE VALUES ON THE X-W GRID.
C
C ************* C
COMMON/XC,XW,X(20),W(17),PQGrel(20,17),PQPM(20,17),
QGMAX(20,17),QGMBMAX(20,17),CM(2)
C
C ************* C
COMMON/XC,XW,X(20),W(17),PQGrel(20,17),PQPM(20,17),
QGMAX(20,17),QGMBMAX(20,17),CM(2)
III=0
100 III=I1+1
IF(XC.XX(III+1)) GOTO 100
III=III+1
200 I=I1+I=1
IF(PQG.WW(IW+1)) GOTO 200
1D=(WW(IW))/W(IW+1)-W(IW)
X1=(X-XX(IW))/XX(IW+1)-XX(IW)
X1=P(QG(IW+1),IW)-PQG(IW,IXW)+X1*X1*1W
A Monte Carlo program for lepton production

\[ X2P = (PQG(IX+1,IW+1) - PQG(IX,IW+1)) \times XD + PQG(IX,IW+1) \]
\[ QG = (X2P - X1P) \times XD + X1P \]
\[ X1P = (PQQB(IX+1,IW) - PQQB(IX,IW)) \times XD + PQQB(IX,IW) \]
\[ X2P = (PQQB(IX+1,IW+1) - PQQB(IX,IW+1)) \times XD + PQQB(IX,IW+1) \]
\[ QQB = (X2P - X1P) \times XD + X1P \]
\[ X1P = (QGHX(IX+1,IW) - QGHX(IX,IW)) \times XD + QGHX(IX,IW) \]
\[ X2P = (QGHX(IX+1,IW+1) - QGHX(IX,IW+1)) \times XD + QGHX(IX,IW+1) \]
\[ QGHX = (X2P - X1P) \times XD + X1P \]
RETURN
END

**Appendix 3: Listing of structure function routines**

**FUNCTION F2(X,Q2)**
\[ F2 = 5.18 \times XV8(X,Q2) + 1.6 \times XV3(X,Q2) + 12.19 \times XS(X,Q2) + 8.9 \times XC(X,Q2) \]
RETURN
END

**FUNCTION XV8(X,Q2)**
\[ S = \text{ALOG(ALOG(0.09)/2.9957)} \]
\[ ETA1 = 0.7 - 0.175 \times S \]
\[ ETA2 = 2.60 + 0.8 \times S \]
\[ XV8 = 3.4 \times ETA1 \times (1 - X) \times ETA2 / (ETA1 + ETA2) \]
RETURN
END

**FUNCTION XV3(X,Q2)**
\[ S = \text{ALOG(ALOG(0.09)/2.9957)} \]
\[ ETA3 = 0.35 - 2 \times S \]
\[ ETA4 = 3.35 + 0.816 \times S \]
\[ XV3 = XV8(X,Q2) - 2 \times ETA3 \times ETA4 \times ETA2 / (ETA3 + ETA4) \]
RETURN
END

**FUNCTION XV(X,Q2)**
\[ XU = 5 \times XV8(X,Q2) \times XV3(X,Q2) \]
RETURN
END

**FUNCTION XD(X,Q2)**
\[ XD = 5 \times XV8(X,Q2) - XV3(X,Q2) \]
RETURN
END

**FUNCTION XUB(X,Q2)**
\[ XUB = XS(X,Q2) \]
RETURN
END
FUNCTION XS(X, Q2)
S = ALOG (ALOG (Q2/0.09/2.9957))
PS = .75 (.169*EXP(-.747*S)*.429-.488*EXP(-.427*S))
& .0282*(.028145*EXP(-.747*S)+1.6335*EXP(-.609*S))
& - .157*EXP(-.667*S)+.25*.009176*EXP(-.667*S))/PS
XS = (PS*(1/XS-1.)*(1-x)**(1/XS-2.))/6.
RETURN
END

FUNCTION XC(X, Q2)
S = ALOG (ALOG (Q2/0.09/2.9957))
PC = .25 (.169*EXP(-.747*S)+.429-.488*EXP(-.427*S)-.11*EXP(-.427*S))
XC = .25 (.0028145*EXP(-.747*S)+1.6335*EXP(-.609*S)-.157*
& - .667*S)+.009176*EXP(-.667*S))/PC
XC = (PC*(1/XC-1.)*(1-x)**(1/XC-2.))/2.
RETURN
END

FUNCTION XG(X, Q2)
S = ALOG (ALOG (Q2/0.09/2.9957))
PG = .51 (.169*EXP(-.747*S))
XG = (.0425397*EXP(-.609*S)-.0090397*EXP(-.1386*S))/PG
XG = PG*(1/XG-1.)*(1-x)**(1/XG-2.)
RETURN
END

C *********************************************************
C C STRUCTURE FUNCTIONS FROM GLUCK, HOFFMANN AND REYA
C DORTMUND PREPRINT DO-TH 80/13, MAY 1980
C C REQUESTED FUNCTION : GAMMA (X)
C C *********************************************************

FUNCTION F2(X, Q2)
F2 = (4.*XU(X, Q2)+XU(X, Q2)+10.*XUB(X, Q2)+2.*XS(X, Q2))
& +8.*XC(X, Q2))/9.
RETURN
END

FUNCTION BETA(X, Y)
BETA = GAMMA(X)*GAMMA(Y)/GAMMA(X+Y)
RETURN
END

FUNCTION XU(X, Q2)
S = ALOG (ALOG (Q2/0.16)/3.2189)
A = .421-.0412*S
C = 2-.6223*S**.8
D = 3.374+.4319*S
XU = 2.*C**A*(1.-X**C)**D/BETA(9+1., A/C)
RETURN
END

FUNCTION XD(X, Q2)
S = ALOG (ALOG (Q2/0.16)/3.2189)
A = -.364-.0368*S
C = 2-.5414*S**.8
D = 5.09+.3463*S
XD = C**A*(1.-X**C)**D/BETA(9+1., A/C)
RETURN
END

FUNCTION XUB(X, Q2)
S = ALOG (ALOG (Q2/0.16)/3.2189)
A = .25+.088*S**.13
B = .8128*S-.2.003*S**1.8+.0831*S**2
C = 3.97*S
D = 7.41.666*S
E = .2487**2.5
F = 27.8+59.66*S
XUB = A*(1.43*X**C**2)*(1.-X)**D + E*EXP(-F*X)
RETURN
END

FUNCTION XU(X, Q2)
S = ALOG (ALOG (Q2/0.16)/3.2189)
A = .0625+.1132*S**.13
B = 12.64*S-.51.70*S*1.8+.38.02*S**2
C = 4.448*S
D = 7.41.562*S
E = .3081*S**2.5
F = 4.26+.67.91*S
XU = A*(1.43*X+C**2)*(1.-X)**D + E*EXP(-F*X)
RETURN
END

FUNCTION XC(X, Q2)
XC = 0.
RETURN
END

FUNCTION XG(X, Q2)
S = ALOG (ALOG (Q2/0.16)/3.2189)
A = .9243+2.31*S**.05
B = 8.538-0.227*S**1.3-.655*S**1.5
C = 53.57-68.79*S**0.3+.19.3*S
D = 6.1+1.454*S
E = 11.29*S**2
F = 4.24+50.71*S
XG = A*(1.43*X+C**2)*(1.-X)**D + E*EXP(-F*X)
RETURN
END

FUNCTION XG(X, Q2)
S = ALOG (ALOG (Q2/0.16)/3.2189)
A = .9243+2.31*S**.05
B = 8.538-0.227*S**1.3-.655*S**1.5
C = 53.57-68.79*S**0.3+.19.3*S
D = 6.1+1.454*S
E = 11.29*S**2
F = 4.24+50.71*S
XG = A*(1.43*X+C**2)*(1.-X)**D + E*EXP(-F*X)
RETURN
END

RETURN
END