Thomas Jefferson National Accelerator Facility

12000 Jefferson Avenue
Newport News, VA 23606

To:	GDH/A$_1^n$/g2 Collaboration	
From:	Xiaochao Zheng	Oct. 22, 2000
Subject:	Instruction for $e^-/\pi/e^+$ Program	
URL:	http://www.jlab.org/~xiaochao/e99117/index.html#data	

Contents:

1. Introduction;
  1.1 e^-, π^- cross section near resonance region – QFS, EPC;
  1.2 e^-, π^- cross section in deep inelastic region – Whitlow, Wiser;
  1.3 e^+ cross section – EPC vs. Wiser;
  1.4 (e, e') elastic tail – Rosetail;

2. Instruction for $e^-/\pi/e^+$ program;
  2.1 List of files;
  2.2 Input file format;
  2.3 Output data files;
  2.4 Reporting to LaTex;
  2.5 Summary;

1. Introduction

Parametrization of inelastic electron scattering (e, e') and electronucleon production (e, N) cross sections for nuclei is a convenient tool for the design of nuclear physics experiments and the analysis of the resulting data. Parametrized cross sections can be used in radiative corrections, assessing the feasibility of a $(e, e'N)$ coincidence measurement and calculating experimental backgrounds.

1.1 e^-, π^- cross section near resonance region – QFS, EPC

Fortran code epc.f and qfs.f[1], developed by J.W. Lightbody and J.S. O'Connell in 1988, are commonly used in predicting doubly differential cross sections for (e, e'), (e, N), and (e, π) in different nuclei.

QFS code is based on five reaction mechanisms, namely (1) quasielastic (e, e') scattering on a bound nucleon, (2) scattering on two interacting nucleons, (3) Δ electroproduction, (4) higher nucleon resonance electroproduction (two resonances centered at $W = 1500$ MeV and 1700 MeV), and (5) deep inelastic (e, e') in the X-scaling regime. According to the original paper, QFS code is able to predict (e, e') cross sections to within 20% for an incident electron in the energy range 0.5-5 GeV and for energy losses greater than 50 MeV. However, QFS code is developed mainly from the existing data in resonance region so it is possible that they are not appliable in deep inelastic region.

For (e,N) and (e,π), **EPC** code was developed based on three reaction mechanisms: (1) quasideuteron mechanism, (2)recoil nucleons following pion production, and (3) quasifree knockout. It is able to predict (e,N) and (e,π) cross sections in the same electron energy range as QFS to within a factor of 2 for nucleon kinetic energies greater than 50 MeV. But at large angles and above an incident electron energy of 1 GeV it underpredict pion yields presumably because of multiple pion production. Again, we are conservative in applying it to deep inelastic region.

There is another code (GPC) for yields of nucleons from bremsstrahlung beams but we are not using it in this software.

1.2 e^-, π^- cross section in deep inelastic region – Whitlow, Wiser

Whitlow code[2] was developed in 1990 by a graduate student working on SLAC E140, a deep inelastic electron-deuterium and electron-iron scattering experiment. The code was generated by combining E140 data with data from all SLAC deep inelastic electron-hydrogen and electron-deuterium cross section measurements between 1970 and 1983 (E49a, E49b, E61, E89a, and E89b). It is commonly used to predict (e,e') cross section in deep inelastic region. We expect that Whitlow code works better than QFS in deep inelastic region, although QFS also include deep inelastic reaction mechanism. The existing Whitlow code is for hydrogen and deuterium targets only. For E99117 we use the sum of these two results for ^3He target.

Wiser code[3] for inclusive photoproduction of both positive and negative protons, kaons, and pions was developed in 1977. It is based on SLAC data using (1) a hydrogen target, (2) a bremsstrahlung photon beam made with electron energies of 5, 7, 9, 11, 15 and 19 GeV, and (3) SLAC 8 GeV/c spectrometer. Because of this specific kinematic regime, Wiser code is commonly used to predict (e,N), (e,K) and (e,π) cross sections in high beam energy, deep inelastic region. Later data show that Wiser code has an overestimation within a factor of 2. In our program, we use result of wiser code for hydrogen target. For ^3He target, we assume photoproductions from bound proton and neutron are the same and there is no correlation effect between nucleons inside ^3He, e.g., we multiply result of wiser code by a factor of 3.

1.3 e^+ cross section – pos_JPChen vs. pos_wiser

We include only one reaction channel, $\pi^0 \to e^+e^-$, for calculating e^+ cross section. The code **POS_JPChen** was first developed by J.P.Chen based on SLAC data and EPC code for yield of π^0. Code **POS_WISER** is developed by using the same reaction channel, but using WISER code for yield of π^0. Consequently, we expect that POS_JPCHEN works good in near resonance region, while POS_WISER works better in deep inelastic region.

1.4 (e, e') elastic tail – Rosetail

Rosetail code computes ratiative tail corrections to electron scattering cross sections. It was first developed in Univ. of Virginia by Rosemary Altulmus and John Wise in 1980. Later several revisions were made and the latest version is based on two theoretical work [4][5]. The program needs the radiation lengths of the pre window, pre target, post window and post target. The possible tail corrections include: (1) the external bremsstrahlung correction, (2) the internal bremsstrahlung correction, which uses either exact calculation from Mo and Tsai[4], or peaking approximation from Stein[5], (3) the multiple-photon correction term can be included.

Before using this program, check if the radiation lengths of the window and target are correct. You can also incorporate other form factors that you need into the program.

[1] J. W. Lightbody, Jr., J. S. O'Connell, **Computers in Physics**, May/June (1988);
[2] L. W. Whitlow, **SLAC-Report-357**, (1990);
[3] D. E. Wiser, **Ph.D. thesis**, Univ. of Wisconsin, (1977);
[4] Mo and Tsai,**Rev. Mod. Phys.**, **41**, 205 (1969);
[5] Stein, et. al. **Phys. Rev. D**,**12**, 1884 (1975).

2. Instruction for $e^-/\pi/e^+$ program

2.1 List of files

First download source code rate.tar from the website, then use

```
tar -xvf rate.tar
```

to decompress the .tar file. Then under current dir, create ./inp and ./dat directory which will be used later.

In rate.tar please find:

(1) Fortran codes for parametrization of the (e, e'), (e, N), (e, π), (e, e^+), cross section:
epc.f;
wiser.f;
qfsrad.f;
rosetail.f;
pos_JPChen.f;
pos_wiser.f;
these fortran codes need an input file under ./inp/ directory. They will generate .dat files under ./dat/ directory.

input: example of the input file needed by the fortran codes above;

(2) C codes for parametrization of the (e, e') cross section:
whitlow.c: c code for whitlow's fit for electron cross section in DIS, it needs an input file "rates.dat" under ./inp/ directory and will generate .dat files under ./dat/ directory.
rates.dat: example for the input file needed by whitlow.c;

(3) csh file for running all codes:
Code run_code_e99117.sh or run_code_testrun.sh are used to simplify the procedure of running all codes listed above. It could copy different input file to ./inp directory, run all codes, then move all output data files to a user-specified directory like '/home/xiaochao/test/code/e99117-newreq/'. It might not be so convenient for people who don't usually use csh files so it is for reference only.

(4) C codes for reporting kinematics, generating summary and Latex file:
readkine.c: This code will extract kinematics from file 'rate.dat', e.g. input file for C code. It will calculate some kinematic variables like Bjorken scaling variable x, invariant mass W, four-momentum transfer q^2, virtual photon polarisation ϵ etc.. The result will be in 'kine.dat' which is also under ./dat directory.

writesum.c: First it asks for how many kinematics you have, then will read cross sections for these kinematics from data files under a specific directory(you can change the data file directory by modifying source code), and generate a summary file ¡summary¿ under the current directory. It will also generate a Latex file which is basically a table which list all the cross sections and π^-/e^-, e^+/e^- ratio from different codes. It should be used after running the csh file above, otherwise it will mess up sth. because it cannot find the correct data files.

2.2 Input file format

(1) input for Fortran codes

All Fortran codes share one input file under ./inp/ directory. Format of 'input' is:

1 2. 1 PI- 5700. 854. 926. 10.0 45. 890.0 130. 20.0 20.0 4774. 4846. 15.00 25.00

where

1: Z of target
2.: A of target
1: N of target
PI-: partile name to be calculated, like PI-, PI+, or PI0
5700.: beam energy in MeV;

854., 926.: electron arm momentum range, min p^e_{min} and max p^e_{max} in MeV/c
10.0: step in electron momentum;
45.0: scattering angle θ in degree;
890.0: electron arm central momentum $p^e_{central}$ in MeV/c
130.0: fermi momentum of target nuclei
20.0: delta separation energy
20.0: bin width
4774., 4846: energy loss min ν_{min} and max ν_{max}, e.g., beam energy minus electron momentum $\nu = E_{beam} - p_e$;
15.00: beam current in microA;
25.00: target length in cm;

(2) input for C codes

Input file for whitlow.c is rate.dat. The format is:

5.7 0.0 0.854 0.01 0.926 45.0 0.0 45.0 1E-22 25.0 50.0

where
5.7: beam energy in GeV
0.0: step in beam energy
0.854: electron momentum in GeV/c, min
0.01: step in electron momentum in GeV/c
0.926: electron momentum in GeV/c, max
45.0: scattering angle in degree, min
0.0: step in scattering angle
45.0: scattering angle in degree, max
1E-22: target density (funny unit, I don't remember, maybe in mol/cm3)
25.0: target length in cm
50.0: beam current in microA

(3) Other information

Other information you need to include: radiation length of the window and target, used by wiser.f, pos_wiser.f, pos_JPChen.f, rosetail.f. In some codes the target radiation length is calculated from the real target length in the input file, you'd better double check it. Then you probablly need to modify the source codes according to your own configurations. You can also modify the source code rosetail.f to incorporate new target form factors.

2.3 Output data file format

For each kinematics(e.g., for each input file), each of these Fortran and C codes will write the results of cross section into two data files. For example, epc.f will generate two data files 'epc.dat' and 'epc_tot.dat'. 'Epc.dat' is a list of electron final momentum(in MeV/c) and the corresponding cross section in unit of (nb/GeV-sr), where the step in electron final momentum is given in the input file. It could be used for estimating how cross section changes within the current momentum setting (here we interpret p_{min} and p_{max} as the momentum acceptance of electron spectrometer). The other output file 'epc_tot.dat' gives the number of points, sum of cross sections of each step and the average cross section for the current momentum setting. This is the source file for writesum.c.

Examples of output file 'epc.dat' and 'epc_tot.dat' are also in the rate.tar.gz:
epc.dat:

x	p_e	$\sigma(nb/GeV-sr)$
0.181811579	528.	0.643409365
0.185617372	538.	0.630819511
0.189438085	548.	0.618768662
0.193273804	558.	0.607224655
0.197124618	568.	0.596157697

epc_tot.dat:

5	3.09637989	0.619275978

2.4 Reporting to LaTEX

If everything goes smoothly the you will have one Latex file after running writerum.c. You can specify the name of Latex file by modifying source code writerum.c, or use the default value '/home/xiaochao/test/memo/memo-testrun/summary_e99117.tex'. Then you can either use this TEX file directly, or copy it to your own report.

Basic commands for LaTEX:

```
>latex filename.tex <return>
>dvips -f filename.dvi > filename.ps <return>
```

2.5 Summary

(1) Decompress rate.tar under current directory;

(2) Check if the radiation lengths are correct, modify the source codes according to your configuration. You can also incorporate your own target form factors in rosetail.f;

(3) Compile all codes;

(4) generate ./inp and ./dat sub-directories;

(5) under ./inp directory, write your own input files 'input' and 'rate.dat'. If you have many kinematic points, generate separate input file for each of them and save them as 'input1',

'input2', etc..;

(6) check csh file run_code_xxxx.sh, make sure all input files and executive files are located at the correct directory. Run csh file by typing 'csh run_code_xxx.sh'. Then check if all output data files are copied and/or moved correctly;

(7) Use writesum to generate summary files in plain text format and/or LaTEX format;

I apologize if you are lost in these codes. Please feel free to contact me if you have any question/comment/suggestion.