Program Version: 3.5
April 24, 2001

MCEEP

Monte Carlo for Electro-Nuclear Coincidence Experiments

P.E. Ulmer
Department of Physics
Old Dominion University
Norfolk, VA 23529

Email: ulmer@jlab.org

Abstract: This manual describes the Monte Carlo computer program, MCEEP, designed to simulate coincidence (e,e$'$X) experiments by averaging theoretical models over an experimental acceptance. MCEEP runs under several unix operating systems (DEC-Ultrix, DEC-Alpha (OSF1), HP-UX, SunOS and Linux (Absoft f77 compiler and GNU g77 compiler)). The program provides for the analysis of measurement uncertainties so that the effects of various systematic errors can be evaluated. This document describes how the program functions and also serves as a user's manual. In addition, simple procedures are given which allow the user to modify the code so that new histograms and/or physics models can be incorporated.

CONTENTS

1 MOTIVATION 4

2 INTRODUCTION 5

3 GENERAL DESCRIPTION OF MCEEP 7
- 3.1 "Event" Generation and Coordinate Systems 7
- 3.2 Sampled Quantities and Jacobians 8
- 3.3 Kinematics 9
- 3.4 Recoil Polarization Observables 10
- 3.5 Spin Precession 14
- 3.6 Target Models 16
  - 3.6.1 Foils 16
  - 3.6.2 LH_2/LD_2 Cryotarget 17
  - 3.6.3 $^{3,4}He$ Cryotarget 17
- 3.7 Layout of the Program 18
- 3.8 Description of Subroutines 20

4 RADIATION/IONIZATION/MULTIPLE SCATTERING 22
- 4.1 Internal Bremsstrahlung 22
  - 4.1.1 Schwinger Correction 23
  - 4.1.2 Radiative Tail 23
    - 4.1.2.1 Elastic Radiative Tail 24
    - 4.1.2.2 Bound State Radiative Tail 25
    - 4.1.2.3 Continuum Radiative Tail 26
    - 4.1.2.4 Multi-Photon Correction 26
- 4.2 External Processes 26
  - 4.2.1 Hadron Energy Loss by Collisions 27
  - 4.2.2 Electron Energy Loss by Collisions 28
  - 4.2.3 Energy Loss Distribution (Straggling) 28
  - 4.2.4 Multiple Scattering though Small Angles 29
  - 4.2.5 External Bremsstrahlung 29

5 OUTPUT OF MCEEP 31
- 5.1 Physics Observables Computed by MCEEP 31
- 5.2 Integrating Over the Experimental Acceptance 32
- 5.3 Histogram Range Determination 34
- 5.4 Histogram Cuts 35
- 5.5 Options for Producing Output 35

6 SPECTROMETER ANALYSIS 36
- 6.1 Analysis of Uncertainties 36
- 6.2 Options 37

7 RUNNING MCEEP 41
7.1 A Sample MCEEP Run 41
7.2 Input file 44
7.2.1 Structure 45
7.2.2 Variable Descriptions 48
7.2.3 Sample Input File 64

8 PHYSICS MODELS OF MCEEP 67
8.1 General Formalism of (e,e′N) 67
8.2 (e,e′N) in PWIA (Options 100 and 200) 67
8.2.1 Factorization of the Cross Section 68
8.2.2 Spectral Functions 68
8.3 Relativistic d(e,e′p)n (Option 300) 78
8.4 Relativisitic PWBA for d(e,e′p)n (Option 350) 78
8.5 Pion Electroproduction in p(e,e′p)π^0 (Option 500) 79
8.6 Pion Electroproduction in p(e,e′π^+)n (Option 600) 79
8.7 Kaon Electroproduction in p(e,e′K^+)Λ (Option 700) 79
8.8 Arenhövel Responses in d(e,e′p)n (Option 804) 80

9 MODIFYING THE PROGRAM 81
9.1 Adding a New Histogram Variable 81
9.2 Incorporating New Physics Models 82

Appendix A: DESCRIPTION OF KINEMATIC VARIABLES 84
Appendix B: MCEEP UTILITIES 106
Appendix C: PORTING MCEEP 107

Chapter 1

MOTIVATION

Coincidence (e,e$'$X) experiments, in which an emerging hadron is detected in coincidence with a scattered electron, can provide information which is not revealed by inclusive (e,e$'$) experiments where only the scattered electron is detected. In addition to elucidating the single-particle character of nuclei, coincidence experiments can, by restricting the final state, distinguish among various aspects of the reaction such as non-nucleonic effects (Meson Exchange Currents and Isobar Configurations for example) and 2-body correlations, *etc.* These experiments are likely to continue to play an important role in our understanding of the nucleus.

Separation of the coincidence cross section into the various electromagnetic response functions can provide very detailed and important information since various aspects of the reaction mechanism are expected to reveal themselves to different extents in these response functions. However, because of the typically large error magnification inherent in determining response functions from measured cross sections, their precise extraction requires the control of systematic errors to a high degree. In particular, at forward electron angles, where many measurements are expected to be performed (to maximize the counting rate as well as the virtual photon longitudinal polarization lever arm in L/T separations) the cross section is extremely sensitive to variations in the electron scattering angle. These extreme variations coupled with the relatively large experimental acceptances needed to attain high statistical precision necessitate an accurate folding of theoretical models over the appropriate acceptance in order to have meaningful comparisons with data. Thus, it is not generally adequate to merely compare a theory evaluated at the central kinematics with acceptance averaged data. With this in mind and with emphasis on general applicability to any coincidence setup the computer code MCEEP (<u>M</u>onte <u>C</u>arlo for (<u>e</u>,<u>e</u>$'$<u>p</u>)) was written. Although, the program (and its name) were initially created with (e,e$'$p) in mind, MCEEP is applicable to any single hadron emission experiment provided the appropriate physics routines are incorporated. In addition, (e,e$'$) elastic scattering is handled; currently models for the proton, deuteron, triton, ^{3}He, ^{4}He and ^{12}C are included.

This document describes the procedure used by MCEEP in computing physical observables based on a variety of physics models. It also serves as a user's manual describing the input file structure, output options and simple prescriptions for adding new histogram variables or physics models.

Chapter 2

INTRODUCTION

MCEEP employs a uniform random sampling method to populate the experimental acceptance. This method differs from a "true" Monte Carlo which generates events distributed according to a predetermined weighting function. Although the latter method more closely resembles the actual experiment and avoids the need to keep track of weights for each event, it becomes unwieldy for the large number of dimensions needed to describe the experiment in its full generality. For example, in addition to the nine variables required to describe the incident beam, scattered electron and emerging nucleon there are other dependences such as the beam polarization and target polarization as well as the implicit dependence of the cross section on the beam-target interaction point. In addition, in contrast to MCEEP's method, the importance sampling technique requires a large number of events in order to obtain precise statistics for small cross sections in the presence of a dominant contribution.

An event is defined as any combination of variables which completely specifies the reaction in the laboratory. For each event, weights which correspond to the relevant observables (cross sections, polarizations, *etc.*) are computed according to a user selected physics model. The histograms represent the accumulation of these weights. Any of the weights corresponding to the various physical observables can be attached to a given histogram by selections made within the user input file.

The program is written in a modular form allowing easy incorporation of specialized subroutines. This is particularly important for the cross section routines so that, with minimal effort, new theoretical models can be introduced as they become available. The kinematics are kept general so that variations with respect to any variable can be examined and experiments with arbitrary geometries (such as out-of-plane with either a raised spectrometer or beam swinger) can be considered. In addition, MCEEP allows for a general 3-D beam-target interaction region so that the effects of extended targets and rastered or defocussed beams can be evaluated.

MCEEP can perform calculations for elastic scattering, (e,e$'$X) to bound states of the residual system or (e,e$'$X) in the (unrestricted) continuum according to the user's choice. For elastic scattering, MCEEP performs a two-dimensional integral over the electron solid angle. The electron final energy as well as the nuclear recoil momentum are all constrained kinematically. For elastic scattering the user may choose to enforce the acceptance cuts of the hadron arm to simulate an A(e,e$'$A) experiment, or leave the recoil kinematically unrestricted (*i.e.* single-arm elastic scattering). For the bound state case, MCEEP performs a five-dimensional integral, wherein the ejectile momentum is calculated from the bound state missing mass (specified in the input file) and the values of the other five randomly selected kinematical variables (the electron momentum and in-plane and out-plane angles and the ejectile in-plane and out-of-plane angles). For the continuum case, the ejectile momentum is also randomly selected and the missing mass is then calculated event-by-event. Of course, bound state scattering can be effectively calculated by choosing the continuum option with a narrowly peaked missing mass distribution function. However, this is usually a very inefficient method since most events will correspond to the tails of the distribution (depending upon how sharply peaked it is) and hence will contribute very little to the integral.

MCEEP also now (Version 3.0 and later) includes radiative effects and more realistic multiple scattering. Both internal and external radiation are handled. Details on these features are given in a separate section.

The program provides for simulation of various detector/spectrometer systems by allowing the user to specify a series of "spectrometer elements" which act on a Transport[1] vector based on the laboratory coordinates directly sampled in the Monte Carlo event loop. Analysis of uncertainties is accomplished by allowing the user to offset a given Transport ray coordinate or to smear it with a gaussian resolution function. Several other types of elements are also available including multiplication of the Transport vector by a matrix, drifts and rotations in field-free regions and spin precession through a series of magnetic elements. In addition, histograms, cuts and Transport vector Ntuples can be defined at any point along the spectrometer line. The cuts can be attached to particular histograms defined at any location or can be applied globally. Note that the so-called "spectrometer" elements are not restricted to magnetic devices but can also be used to simulate simple detector systems, such as for neutron detection experiments, *etc.*

Output of MCEEP consists of a set of histograms representing dynamical quantities (polarizations, cross sections or the actual yield) as a function of various kinematical variables. The physics observable as well as the kinematic variable are selected by the user via an input file. In addition, the user can specify a set of cuts to be applied to either all histograms (global cuts) or to individual histograms (specific cuts). Histograms of Transport ray coordinates can also be requested in the spectrometer analysis section of the input file. MCEEP also provides for Ntuple output (for both kinematics variables and Transport variables) which can be converted by an external program (*mceep_util*) to a PAW-compatible[2] HBOOK[3] file. Finally, MCEEP produces a summary file which consists of a formatted display of the user input, histogram summary tables and various statistics which reflect the population of the kinematical variables.

In the next chapter the general layout of the program as well as the main subroutines employed by MCEEP are described. The remaining chapters discuss various aspects of the program in detail.

Chapter 3

GENERAL DESCRIPTION OF MCEEP

3.1 "Event" Generation and Coordinate Systems

An event is chosen by randomly selecting a momentum and two orientation angles for each particle (the scattered electron and emerging hadron) somewhere within the experimental acceptance. In the most general case, the beam-target interaction point is also chosen at random somewhere within the target volume (consistent, of course, with the beam profile). The angular apertures of the spectrometers can be either rectangular or elliptical. The angles are actually chosen by specifying *spatial* coordinates (one along X and one along Y where these coordinates are defined according to the Transport notation) within each of the two apertures. This combined with the beam-target interaction coordinates specify the actual orientation angles relative to a fixed system in the laboratory for each of the two emerging particles. This method of sampling is very efficient since it guarantees that the particles lie within the apertures even for extended targets. In contrast, if one selects *angles* at random (relative to a fixed coordinate system in the laboratory) there is no guarantee that for any origination point within the target the resulting particle will lie within the solid angle defining aperture. However, the former method has one drawback, namely the need for Jacobians relating the sampled "spatial" coordinates to the actual angles in the laboratory.[4] These Jacobians will be described below. Finally, the user may elect to give the incident beam a spread in energy or angle in which case these quantities are chosen randomly according to a prescribed (gaussian) distribution with mean values equal to the nominal input values.

All quantities are specified relative to one fixed coordinate system. Having a fixed coordinate system simplifies the calculation of various kinematical quantities through vector operations. The system is defined as the "**laboratory**" coordinate system. Thus, the nominal beam orientation defines the Z axis (in the case of a beam swinger "nominal" means the horizontal projection of the beam direction - *i.e.* in the plane of the laboratory floor). The Y axis points vertically upward (relative to the laboratory floor) and $\hat{x} = \hat{y} \times \hat{z}$.

From the randomly selected coordinates (hereafter referred to as the "laboratory coordinates" since they refer to a fixed coordinate system in the laboratory), the coordinates in the "**Transport**" system can be computed. This is convenient for incorporating the properties of spectrometers or detection systems into the analysis. For example, it may be of interest to examine the distribution of events at the focal plane of a spectrometer since this has direct relevance for the detectors. Also, as will be discussed in detail later, the effects of uncertainties and errors can be evaluated by introducing offsets and smearing into the appropriate Transport coordinate. The method of incorporating detection systems and various measurement uncertainties involves starting with the Transport vector at the target for both arms and applying functional mappings (currently only multiplication by a 1st or 2nd order "matrix" is supported), offsets and resolution smearing to this coordinate vector. The final element in the spectrometer analysis input (see the chapter on spectrometer analysis for more details) must convert the resulting coordinate vector back to the Transport vector at the target. Thus, a closed loop is formed starting and ending at the target Transport vector. From this "affected" Transport vector the laboratory coordinates can again be determined. From the laboratory coordinates, "physical variables" which are more convenient for the calculation of dynamical quantities are computed. The current version of MCEEP also allows for "turning off" the spectrometer line in computing kinematic variables. This has the advantage that spectrometer/detector distributions can be examined without affecting the generation of the kinematic quantities. (For example, inaccuracies resulting from differences between the forward and reverse maps are avoided.)

3.2 Sampled Quantities and Jacobians

For the most general case of scattering to the continuum of the $A-1$ nuclear system and radiating a hard photon, a MCEEP event is essentially defined by the sampling of seven coordinates. All but the radiated photon energy are sampled uniformly (the photon energy is sampled with a $1/k$ distribution). The sampled coordinates are (see below for definition of the standard coordinate system):

- ϕ_{en}: The "nominal" in-plane angle of the electron.
- θ_{en}: The "nominal" out-of-plane angle of the electron.
- p_e: The electron momentum.
- ϕ_{pn}: The "nominal" in-plane angle of the hadron.
- θ_{pn}: The "nominal" out-of-plane angle of the hadron.
- p_p: The hadron momentum.
- k: The energy of the radiated photon.

For elastic scattering, only ϕ_{en} and θ_{en} are sampled. All other quantities can be calculated. (For internal bremsstrahlung, the photon energy is also sampled.) For the bound state problem, p_p is determined from the other sampled quantities (including possibly the radiated photon energy).

"Nominal" implies that the angles are picked with reference to the target center. The "actual" angles in the laboratory (*i.e.* the angles in which the cross section is differential) are then calculated from the corresponding point within the solid angle defining aperture (which depends on the drift distance specified in the input file) and the beam-target interaction vertex. The actual angles then specify the direction of the particle defined by these two points. Since the cross section is differential in the actual angles, whereas the nominal angles are uniformly sampled, we need to multiply the cross section (and phase space) by a Jacobian for each solid angle.

For each particle, the solid angle differential is given by

$$d\Omega_a = \cos\theta_a d\theta_a d\phi_a \tag{1}$$

where ϕ_a and θ_a are the "actual" in-plane and out-of-plane angles of the particle referred to the standard laboratory coordinate system. Note that $\cos\theta_a$ is involved instead of the usual $\sin\theta_a$ since MCEEP defines θ as the out-of-plane angle (which is $90^\circ - \theta_{polar}$). For uniform sampling of $d\Omega_a$, we want uniform sampling of $d\theta_a$ and $\cos\theta_a d\phi_a$. Instead, MCEEP uniformly samples $d\Omega_n$, where

$$d\Omega_n = \cos\theta_n d\theta_n d\phi_n. \tag{2}$$

So our Jacobian is:

$$\mathcal{J} = \frac{d\Omega_a}{d\Omega_n} = \frac{\cos\theta_a}{\cos\theta_n}\left(\frac{\partial\theta_a}{\partial\theta_n}\frac{\partial\phi_a}{\partial\phi_n} - \frac{\partial\theta_a}{\partial\phi_n}\frac{\partial\phi_a}{\partial\theta_n}\right). \tag{3}$$

Evaluation of the derivatives requires relations between (ϕ_a, θ_a) and (ϕ_n, θ_n).

Let's define three vectors relative to the standard laboratory coordinate system, $\vec{r}_0 = (x_0, y_0, z_0)$, $\vec{r} = (x, y, z)$ and $\vec{r}_a = (x_a, y_a, z_a)$, where $\vec{r}_0$ is the vector from the origin to the vertex, $\vec{r}$ is the vector from the vertex to the intersection of the outgoing particle with the aperture plane (point P) and $\vec{r}_a$ is the vector from the origin to point P. Then $\vec{r} = \vec{r}_a - \vec{r}_0$. Since r_a is defined in terms of the nominal angles, we have:

$$\begin{aligned} x &= d_0\cos\theta_n\sin\phi_n - x_0 \\ y &= d_0\sin\theta_n - y_0 \\ z &= d_0\cos\theta_n\cos\phi_n - z_0 \end{aligned} \tag{4}$$

where d_0 is the nominal drift distance (*i.e.* the distance between the origin and point P). Similarly, in terms of the actual drift distance, d (*i.e.* the length of the vector $\vec{r}$):

$$\begin{aligned} x &= d\cos\theta_a\sin\phi_a \\ y &= d\sin\theta_a \\ z &= d\cos\theta_a\cos\phi_a. \end{aligned} \tag{5}$$

Therefore, we have:

$$\begin{aligned} \sin\theta_a &= \frac{d_0\sin\theta_n - y_0}{d} \\ \cos\theta_a\cos\phi_a &= \frac{d_0\cos\theta_n\cos\phi_n - z_0}{d} \end{aligned} \tag{6}$$

where $d^2 = x^2 + y^2 + z^2$. These relations are sufficient to determine the solid angle Jacobian, though the simple but somewhat gorey details are omitted here.

3.3 Kinematics

For the general $A(e, e'X)B$ reaction we define the various 4-momenta as:

$e_\mu = (e, \vec{e})$	incident electron 4-momentum
$e'_\mu = (e', \vec{e}')$	scattered electron 4-momentum
$q_\mu = (\omega, \vec{q})$	4-momentum transfer
$p_\mu = (E, \vec{p})$	"ejectile" 4-momentum
$P_\mu = (M_T, \vec{0})$	target nucleus 4-momentum
$P'_\mu = (E_r, \vec{p}_r)$	recoil nucleus 4-momentum

Then $q_\mu = e_\mu - e'_\mu$ and $Q^2 = -q_\mu^2 = |\vec{q}\,|^2 - \omega^2$. The missing mass is given by

$$\epsilon_m = \omega + M_T - E \tag{7}$$

for elastic scattering and by

$$\epsilon_m = \sqrt{(\omega + M_T - E)^2 - p_r^2} - M_T + m_X \tag{8}$$

otherwise. Here m_X is the ejectile mass and the radical represents the mass of the recoiling nucleus. For the bound state problem, the ejectile momentum is determined from the randomly sampled quantities and the (fixed) missing mass from the solution of a quadratic equation:

$$a|\vec{p}\,|^2 + b|\vec{p}\,| + c = 0 \tag{9}$$

where

$$\begin{aligned} a &= 4(\alpha - Q^2 \cos^2\theta_{pq}) \\ b &= 4|\vec{q}\,|\beta \cos\theta_{pq} \\ c &= 4\alpha\, m_X^2 - \beta^2 \end{aligned} \tag{10}$$

with

$$\begin{aligned} \alpha &= (\omega + M_T)^2 \\ \beta &= Q^2 - \alpha + (\epsilon_m + M_T - m_X)^2 - m_X^2. \end{aligned} \tag{11}$$

Here, θ_{pq} is the angle between the ejectile and momentum transfer directions. In some cases, both roots yield physically acceptable solutions. In these cases, one of the two roots is randomly selected and the event carries a weighting factor of two to account for the fact that each root is selected only half the time.

For the bound state problem, the missing mass, ϵ_m, is one of the input variables. This variable (Fortran name: *EM_BOUND*) should be assigned a value consistent with the above equations for ϵ_m, depending on the particular reaction. Examples are:

$EM\_BOUND =$ excitation energy of B	for $A(e,e'N)B$
$EM\_BOUND = 0$	for $A(e,e'A)$ elastic
$EM\_BOUND = m_\pi$	for $p(e,e'p)\pi$
$EM\_BOUND = m_\Lambda + m_K - m_p$	for $p(e,e'K)\Lambda$

3.4 Recoil Polarization Observables

In addition to cross sections, MCEEP provides histograms of recoil polarization observables. The three components of the emerging hadron's polarization can be calculated using various theoretical models. Since the polarizations are computed relative to the reaction plane (*i.e.* the plane containing $\vec{q}$ and the emerging hadron) the orientation of which changes from event to event, the resulting histograms represent an average of the polarization components over a range of orientations in the laboratory. In fact, in parallel kinematics, the orientation of the reaction plane sweeps out the full 2π angular range as the direction of the emerging hadron describes a cone about $\vec{q}$. Thus, for small θ_{pq}, polarization components perpendicular to $\vec{q}$ will average to zero. It is therefore necessary to define a coordinate system which is fixed in the laboratory to which the polarization components can be referred. A coordinate system tied to the hadron spectrometer is most convenient since one can then compute effects of the spectrometer on these components (from precession of the spin through the spectrometer magnetic elements) and can examine the distribution of events in a polarimeter which is also fixed relative to the spectrometer.

We define three right-handed coordinate systems, L, R and S, all of which are at rest with respect to the laboratory, although R is event-dependent. These systems will now be defined.

"L" denotes the usual "laboratory" frame (defined earlier). The Z axis points along the nominal beam direction (in the plane of the laboratory floor), the Y axis points vertically upward and $\hat{x} = \hat{y} \times \hat{z}$. We define three unit vectors, $\hat{l}_1$, $\hat{l}_2$ and $\hat{l}_3$ pointing along the X, Y and Z "laboratory" axes respectively.

"R" denotes the "reaction" frame. This system is the preferred frame for the calculation of the polarization observables. Here, the Z axis points along the hadron momentum, the X axis is normal to the reaction plane (*i.e.* the plane containing $\vec{q}$ and the hadron momentum) and the Y axis direction is given by $\hat{z} \times \hat{x}$. By convention, the X axis points upward for coplanar kinematics with $\phi_x = 0$. (Note that $\phi_x = 0$ corresponds to hadrons forward of $\vec{q}$.) We then define three "reaction" frame unit vectors, $\hat{r}_1$, $\hat{r}_2$ and $\hat{r}_3$, corresponding to the usual n (normal), t (transverse) and l (longitudinal) polarization components respectively (as shown in Figure 1). Note that with these conventions, $\hat{r}_1$ and $\hat{r}_3$ point along the X and Z coordinate axes respectively, whereas $\hat{r}_2$ points in the $-Y$ direction. Thus, although the X, Y and Z coordinate axes define a right-handed system, the three unit vectors, $\hat{r}_1$, $\hat{r}_2$ and $\hat{r}_3$ form a left-handed system. The polarization vector, referred to R can be constructed as:

$$\vec{P} = p_n \hat{r}_1 + p_t \hat{r}_2 + p_l \hat{r}_3. \tag{12}$$

With the above conventions:

$$(R \text{ frame vectors}) \begin{cases} \hat{r}_1 = \vec{q} \times \vec{p} / |\vec{q} \times \vec{p}\,| \\ \hat{r}_2 = \hat{r}_1 \times \hat{r}_3 \\ \hat{r}_3 = \vec{p}/|\vec{p}\,|. \end{cases} \tag{13}$$

We wish to determine the three polarization components with respect to unit vectors in the S frame. "S" denotes the "spectrometer" frame, which is fixed relative to the laboratory for a given spectrometer setting (as opposed to R which depends on the scattering event). This frame is the preferred frame for examination of those quantities directly measured by the polarimeter. Here, the Z axis points along the spectrometer optic axis from the target, the Y axis is normal to the spectrometer midplane and the X axis is in the spectrometer midplane normal to Z. By convention the X axis points downward for a spectrometer which bends vertically upward (*i.e.* along the direction of increasing dispersion) and $\hat{x} = \hat{y} \times \hat{z}$. We define three unit vectors in the "spectrometer" system, $\hat{s}_1$, $\hat{s}_2$ and $\hat{s}_3$ which point along the X, Y and Z coordinate axes respectively. These three unit vectors thus define the orientation of the polarization vector at the target relative to the spectrometer. In terms of the spectrometer angles, θ and ϕ (where θ is the out-of-plane angle and ϕ is the azimuthal angle relative to the beam; note

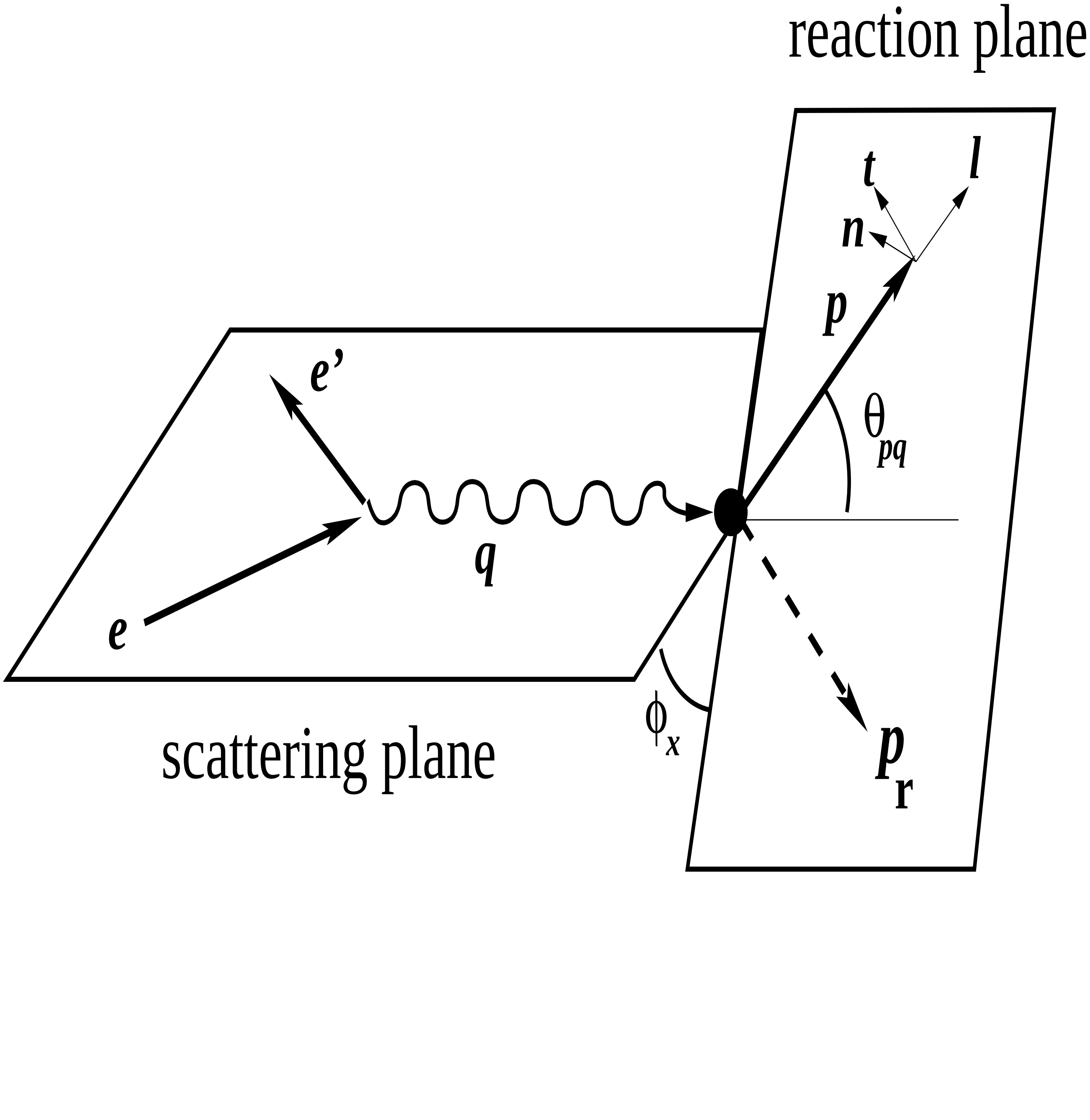

Figure 1. *Kinematics of (e,e′p) including the coordinate system for the "reaction-frame" polarization observables.*

that these angles are not the usual spherical coordinate angles due to the definition of the "laboratory" coordinate system) we can find the components of each unit vector relative to the L frame. By applying three successive passive rotations the "laboratory" to "spectrometer" rotation matrix can be obtained:

$$\mathcal{R}_{L\to S} = \mathcal{R}_Z(\theta_{bp})\mathcal{R}_X(-\theta)\mathcal{R}_Y(\phi) \tag{14}$$

where the subscripts on each matrix on the right-hand side specify the axis of rotation and where positive rotation angles correspond to a coordinate rotation with a positive sense as given by the right hand rule. Here, θ_{bp} defines the orientation of the spectrometer bend plane. For a spectrometer which bends vertically upward, $\theta_{bp} = -90°$, and $\theta_{bp} = 0$ for a spectrometer which bends rightward, *etc.* Note, $\phi > 0$ corresponds to a spectrometer on the left side of the beam (looking toward the beam dump) and $\theta > 0$ for a spectrometer raised above the floor. To minimize confusion, the explicit forms of these rotation matrices are:

$$\mathcal{R}_Y(\phi) = \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}, \tag{15}$$

$$\mathcal{R}_X(-\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \tag{16}$$

and

$$\mathcal{R}_Z(\theta_{bp}) = \begin{pmatrix} \cos\theta_{bp} & \sin\theta_{bp} & 0 \\ -\sin\theta_{bp} & \cos\theta_{bp} & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{17}$$

Now, let r_{ij} be the i^{th} component of the j^{th} R frame unit vector referred to the laboratory system, L. Further, let s_{ij} be the i^{th} component of the j^{th} S frame unit vector referred to L. The components of the polarization vector in the L frame are then

$$p_i^L = r_{ij}p_j^R \tag{18}$$

where $j = 1, 2, 3$ for p_n, p_t and p_l respectively and a sum over repeated indices is implied. Thus, the rotation matrix r_{ij} gives the components of the polarization vector in L from the components in R. We can then determine the components of $\vec{P}$ in the spectrometer frame by taking the dot product of $\vec{P}$, as given by its components above, with the appropriate spectrometer unit vector also expressed in terms of its components in L:

$$p_k^S = s_{ik}p_i^L = s_{ik}r_{ij}p_j^R \tag{19}$$

with s_{ik} as given above in terms of θ, ϕ and θ_{bp}. Effectively, the matrix s_{ij} gives the components in L from the components in S. We can define a matrix l_{ij} which is the inverse of s_{ij} (or the transpose in this case: $l_{ij} = s_{ji}$) so that:

$$p_k^S = l_{ki}r_{ij}p_j^R. \tag{20}$$

Thus, r_{ij} gives components in L from components in R and l_{ij} gives components in S from components in L. One need only take the matrix product to go from R to S.

3.5 Spin Precession

[This section contains many contributions from Mark Jones.]

MCEEP allows for spin precession using either a thin-lens model or via a COSY spin matrix file. In the latter case, the user can supply the file or can use standard files generated for the JLAB-Hall A spectrometers for proton ejectiles.

The thin-lens model assumes that the precession angle, χ, is related to the net bend angle, θ_b, as:

$$\chi = \frac{g-2}{2}\,\gamma\,\theta_b \tag{21}$$

where γ is the ejectile's Lorentz factor ($= E/m$) and g its gyromagnetic ratio. The bend angle is computed from the initial and final Transport vectors for a given magnetic element. Further, the rotation axis for the spin is assumed to be along $\vec{k}_i \times \vec{k}_f$ where $\vec{k}_i$ and $\vec{k}_f$ are the ejectile momentum vectors corresponding to the initial and final Transport vectors. This, effectively, gives the spin precession matrix, S_{ij}.

In the case of COSY, the matrix S_{ij} is obtained from the supplied COSY matrix file. In either case (thin-lens or COSY), the resulting spin precession matrix gives the final polarization direction, $\hat{p}'$, (here the superscript "S" denoting that the vector is referred to the (local) spectrometer system has been dropped for brevity of notation) from the initial polarization direction, $\hat{p}$. In terms of vector components:

$$\hat{p}'_i = S_{ij}\hat{p}_j. \tag{22}$$

In MCEEP the values of the indices correspond to n, t and l respectively for $i, j = 1, 2, 3$. For a vertically bending spectrometer, the matrix takes the form

$$\begin{pmatrix} S_{nn} \approx \cos\chi & S_{nt} \approx 0 & S_{nl} \approx \sin\chi \\ S_{tn} \approx 0 & S_{tt} \approx 1 & S_{tl} \approx 0 \\ S_{ln} \approx -\sin\chi & S_{lt} \approx 0 & S_{ll} \approx \cos\chi \end{pmatrix} \tag{23}$$

where the values after the $\approx$ sign correspond to bending in a vertical plane and χ is as defined above. In general, the spin matrix elements depend on x, theta (dx/dz), y, phi (dy/dz) and momentum and so each event will have a distinct spin matrix.

In Figure 2, the distribution for each spin matrix element is given for protons (of central momentum 1378 MeV/c) from pion production in $p(\vec{e}, e'\vec{p})\pi^0$. The y-axis is the yield from this reaction. For a 1378 MeV/c proton and the JLAB-Hall A hadron spectrometer with its 45° bend angle, $\chi = 143°$. One can see that the distributions are centered around the expected simple dipole magnet values for the spin matrix elements. S_{nt}, S_{tn}, S_{tl} and S_{lt} are centered around zero. S_{tt} has almost no variation from unity. S_{nl} is centered around 0.6 and S_{ln} around -0.6. S_{nn} and S_{ll} should peak at -0.80, but with the crowding near -1, one needs to plot the distribution of $\arccos(S_{nn})$ to see the peak.

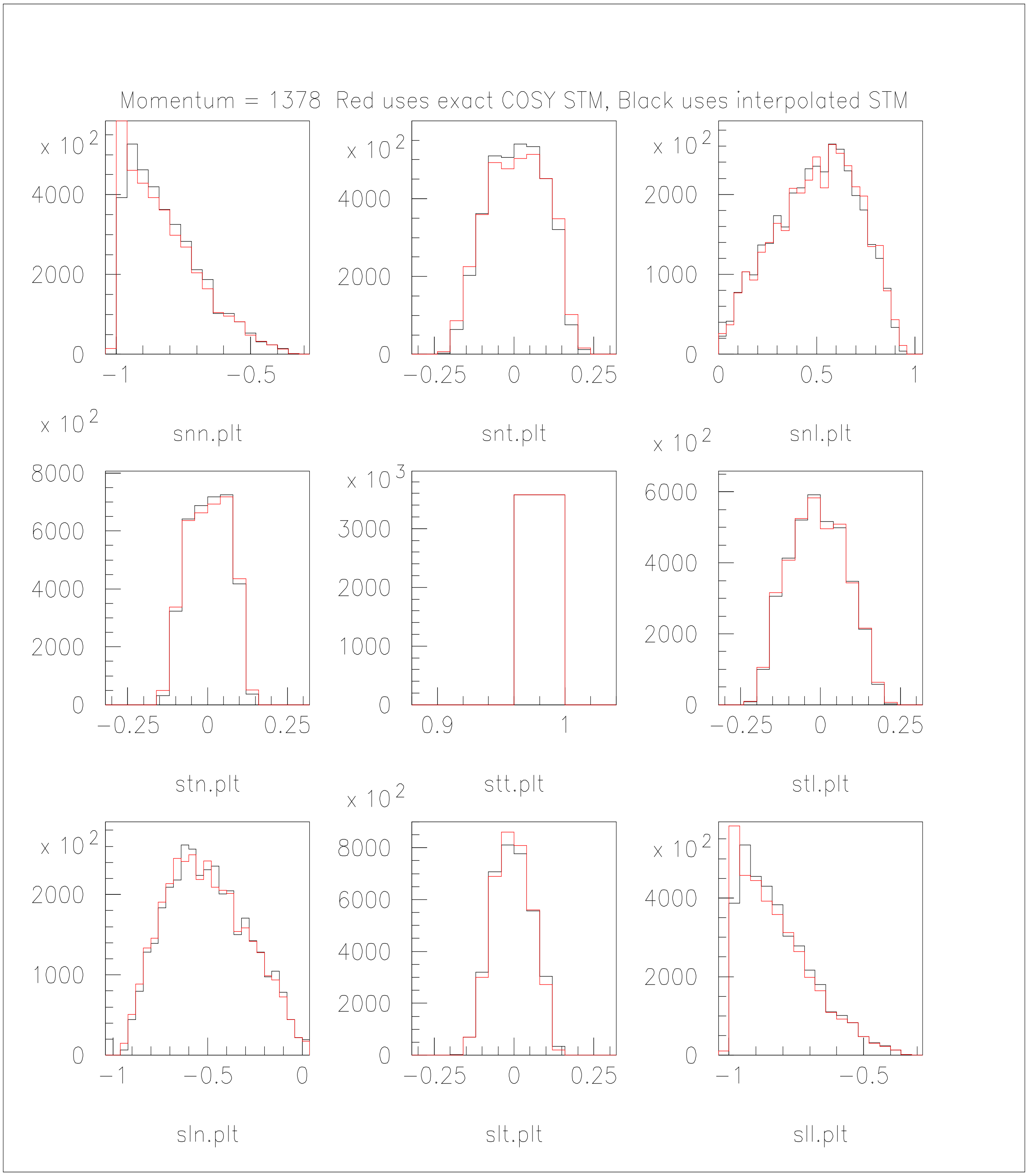

Figure 2. *Spin precession matrix elements for $p(\vec{e}, e'\vec{p})\pi^0$ with the JLAB-Hall A spectrometers and a central proton momentum of 1378 MeV/c. See the text for details.*

3.6 Target Models

MCEEP now supports three different target models: foil(s), the JLAB Hall A LH_2/LD_2 cryotarget cell and the JLAB Hall A ^{3,4}He cryotarget cell ("tuna can"). These are described in detail below.

For a description of the target parameters see the section on the input file. The target length is only important for geometry and for the energy loss/multiple scattering calculations; it does not affect the luminosity. The luminosity is entered as a separate input parameter.

3.6.1 Foils

For targets consisting of either a single foil or multiple foils, the target model parameter ($ITARG\_MOD$) must be set to one. The foils can have any horizontal angle relative to the beam although no provision has been made for vertical tilts (*i.e.* rotations about a horizontal axis). If the energy loss calculation is turned on (in the user input file) the path lengths for the beam and outgoing particles are calculated. The following assumptions have been made:

- All foils have the same orientation (*i.e.* the same horizontal angle). It would be relatively easy to relax this requirement if there is a need.
- All foils have their surfaces vertical.
- A horizontal angle of zero implies the foils are normal to the nominal beam direction.
- Horizontal angles for the foils should be between -90° and 90°.
- The horizontal angle (PHI_TARG) has the same convention as for the spectrometer ϕ angles, except for a 90° relative rotation between the respective $\phi = 0$ values. That is, $\phi = 0$ for the spectrometers imply they are along the beam, whereas $\phi = 0$ for the target implies the foil is normal to the beam. The sense of rotation is defined in the same way for both.
- The outgoing particles never hit foils other than the foil with the primary scattering vertex.
- The starting and ending positions of each foil ($TARG\_LO(J)$ and $TARG\_HI(J)$) as specified in the input file already account for any nonzero target angle.
- The geometry is two-dimensional, ignoring the particles' out-of-plane angles.
- The spectrometer central angle is used to calculate path lengths. This allows the path length to be calculated before the particle angles are selected.

The path length for the beam includes all target material up to the vertex. For the outgoing particles, only the foil with the vertex is assumed to contribute. Depending on the spectrometer and foil angles, an outgoing particle may exit through either the front (downstream) or back (upstream) faces. The relative interaction point is chosen randomly from the cumulative target thickness. Once this point is chosen, the particular foil with the vertex is determined and also the absolute vertex position. For a given foil, let's assume that the vertex occurs a distance D from the back face and a distance d from the front face. (These distances are measured along the beam, so that $D + d = T(i)/\cos\phi_t$ where $T(i)$ is the actual thickness of the relevant foil and the division by $\cos\phi_t$ accounts for the foil's orientation.) Then we have for the path length traversed by the outgoing particle for each of the two cases:

$$
\begin{aligned}
l_{front} &= \frac{d\cos\phi_t}{\cos(\phi_s - \phi_t)} \qquad \text{for } |\phi_s - \phi_t| \leq 90^\circ \\
l_{back} &= \frac{-D\cos\phi_t}{\cos(\phi_s - \phi_t)} \qquad \text{for } |\phi_s - \phi_t| > 90^\circ
\end{aligned}
\tag{24}
$$

where ϕ_s is the spectrometer central angle.

3.6.2 LH_2/LD_2 Cryotarget

Here *ITARG_MOD* must be set to two. A cylindrical geometry is assumed for the cell with the beam coinciding with the cylinder's axis. The cell radius is taken to be 0.0318 m. The length of the cell is specified in the user input file. Scattering from the various walls and windows is taken into account (provided the energy loss calculation is turned on in the input file). Provision is made for outgoing particles to exit either through the cell endcap or the sidewall, depending on the kinematics for a given event.

3.6.3 ^{3,4}He Cryotarget

Here *ITARG_MOD* must be set to three. A cylindrical geometry is assumed for the cell but now with the beam perpendicular to the cylinder's axis. The beam is assumed to pass through the center of the cell. The cell radius is taken to be 0.0516 m. The user should make certain that the length of the cell specified in the input file corresponds to this radius. Scattering from the various walls and windows is taken into account (provided the energy loss calculation is turned on in the input file). Presently, the effective wall thickness seen by the particles assumes scattering at the cell's center, and hence is a constant.

3.7 Layout of the Program

A block diagram of the program is shown in Figure 3. Subroutines are shown to the right of the double arrow and are called in order reading down the page as indicated by the double arrow. Only the main subroutines and those relevant to the present discussion are shown. Subroutines within the main event loop are indented.

At the start of the Monte Carlo event loop, the "laboratory" coordinates, q_i, for both the scattered electron and the ejectile are chosen randomly. From these coordinates a Transport vector at the target, x_i^t, is constructed for each arm by subroutine LAB_TO_TRPT. Various operations can then be applied to these Transport vectors via the subroutine SPECTROMETER. From the resulting "affected" Transport vectors, $x_i^{t\prime}$, new laboratory coordinates, q_i', are calculated by the subroutine TRPT_TO_LAB. All derived kinematical quantities are then calculated from these variables by subroutine KINEM. Note that for this to make sense, the "affected" Transport vectors returned by SPECTROMETER should correspond to the target position. Thus, for example, any Transport matrices applied within SPECTROMETER, should be eventually "undone" by application of their corresponding inverses. This can be avoided by specifying (in the input file) that the spectrometer operations are not to affect the final kinematic quantities.

As indicated in Figure 3, there is some apparent repetition within the main event loop in that several subroutines are called twice. This is because subroutine SPECTROMETER affects the kinematics at two levels. As is explained in detail below, some of the spectrometer elements affect only the cross sections while others affect only the kinematic bins for the histograms. This is necessary to account for various types of measurement errors. For instance, spectrometer misalignments do not affect the histogram bins into which the events are placed since these errors are presumably unknown. Rather each bin is effectively weighted by a cross section which reflects the actual kinematics. On the contrary, multiple scattering effects the kinematic bins but not the cross sections since this secondary scattering (for the *scattered* electron and ejectile) occurs, by definition, **after** the primary (e,e$'$X) scattering. Thus, in the first call to SPECTROMETER only those operations which affect the cross sections are applied. After the cross sections are computed by subroutine PHYSICS, the kinematics are reset to their original (*i.e.* before the call to SPECTROMETER) values. Then SPECTROMETER is called and those operations affecting the kinematic bins are applied. The histograms are updated after the second call to SPECTROMETER.

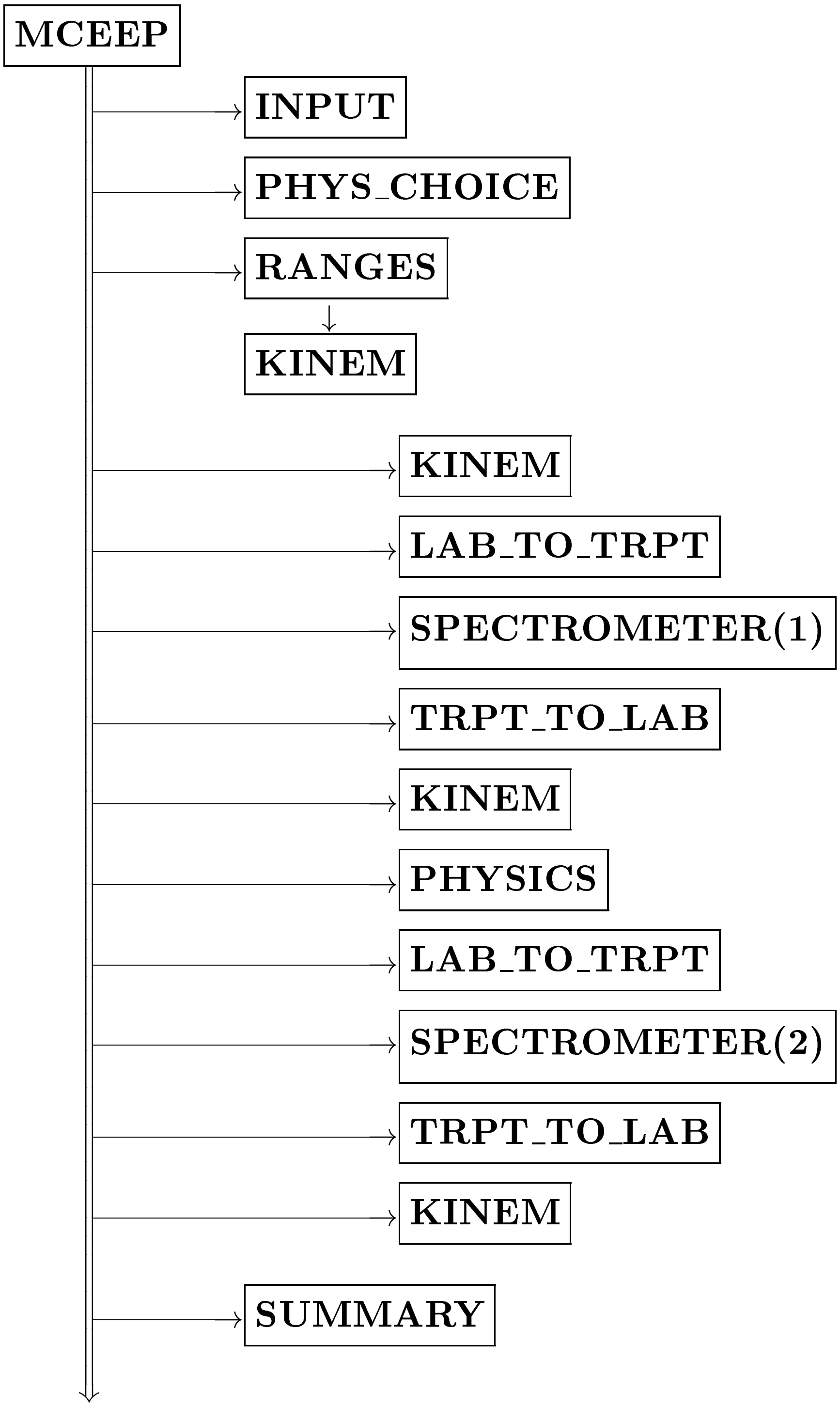

Figure 3 *Block diagram of the main subroutines called by MCEEP. Subroutines are to the right of the double arrow and are called in order reading down the page. Indented subroutines are within the main Monte Carlo event loop. For clarity, only the main subroutines are shown.*

3.8 Description of Subroutines

The subroutines are now briefly described in the order in which they are called by the main program.

INPUT All input is obtained from the user input file by the subroutine INPUT. All input variables are placed into COMMON blocks within the Fortran INCLUDE file *input.cmn* (located in mceep/cmn) so that the main routine has access to them. A complete description of the input variables is given in this document as well as in the comment header within INPUT itself.

PHYS_CHOICE The physics model is chosen within the subroutine PHYS_CHOICE which is a menu-driven routine. This routine is called only once within MCEEP. PHYS_CHOICE, in addition to obtaining the user's physics selection, performs any preliminaries such as opening and reading files. Also, if one of the factorized (e,e′N) options is chosen, PHYS_CHOICE will call subroutine SPECT_SETUP which allows the user to select a spectral function from a menu and performs any necessary preliminaries.

RANGES This subroutine is a preprocessor which provides default ranges for all histograms (other than spectrometer analysis (*i.e.* Transport) histograms). RANGES is described in more detail in the chapter on histograms.

KINEM KINEM performs calculations of all kinematic variables so that they may be histogrammed. Each variable is equivalenced to an element of the array VAR within the Fortran INCLUDE file *var.cmn* (located in mceep/cmn). KINEM is called several times within the program. The first call is initiated by subroutine RANGES so that the default limits for all histogram variables can be determined. KINEM is also called three times within the main event loop. The first call merely calculates the ejectile momentum if the bound state option is selected. For the continuum option, the ejectile momentum is one of the randomly selected variables, whereas for the bound state option this value is calculated from the other randomly selected variables and the specified missing mass. The remaining two calls determine all the kinematic variables, first for the evaluation of cross sections and next for the determination of the histogram channels in which to place the event. The reason for this is explained above and in more detail in the chapter on spectrometer analysis. More details about the kinematic variables are given in the section on adding new histogram variables.

LAB_TO_TRPT This subroutine returns a Transport vector for a given arm based on the "laboratory" coordinates which are passed to it.

SPECTROMETER This subroutine performs various operations on the Transport vectors as specified in the user input file. SPECTROMETER also provides for histograms of any Transport coordinate (or Ntuples of the Transport vector) anywhere along the spectrometer line. The various options of SPECTROMETER are described in detail later.

TRPT_TO_LAB This subroutine returns the "laboratory" coordinates based on the Transport vector passed to it (and is thus the inverse of subroutine LAB_TO_TRPT). Subroutine KINEM determines all derived kinematical quantities based on these "laboratory" coordinates.

PHYSICS This subroutine calculates cross sections and polarizations according to the selection made via PHYS_CHOICE.

SUMMARY SUMMARY produces a formatted output of various statistics along with a histogram summary and the input contained within the user input file. This information is written to the file *[prefix].sum* where "*[prefix]*" is the same file prefix as for the MCEEP input file.

Chapter 4

RADIATION/IONIZATION/MULTIPLE SCATTERING

In addition to the reaction of interest, charged particles experience other interactions which must be taken into account. These effects can be broadly classified as internal and external, depending on whether the interaction occurs on the same atom as for the primary reaction, or some other atom.

The internal effects (internal bremsstrahlung) give rise to the radiative tail and the Schwinger correction. The user can elect to either include these effects or turn them off for testing purposes or for rough estimates. This choice is independent of the calculation of the external effects (which can also be turned off or on).

The external effects manifest themselves in two ways: a loss of energy by the particle and a deflection of the particle from its incident direction. There are many interactions per unit path length through a material and the cumulative effect of these interactions accounts for the observed particle energies and angles. These processes and interactions are: collisions with the atomic electrons of the material, elastic scattering from nuclei, emission of Čerenkov radiation and external bremsstrahlung. **At present, neutral ejectiles are not treated.**

4.1 Internal Bremsstrahlung

Internal bremsstrahlung corresponds to radiation emitted during the primary nuclear interaction. For now, radiation by the hadronic system is ignored. The internal radiation can be further subdivided according to whether the radiated photon is real or virtual (*i.e.* reabsorbed). Further, real photons can be classified according to whether their energies are above a cutoff energy (hard photons) or below the cutoff (soft photons). The soft photon contribution combined with the virtual photon (and vacuum polarization) contribution are collectively referred to here as the "schwinger" correction.[5] Hard photon emission leads to the so-called radiative tail.

In order to allow calculation of the radiative tail **and** the "peak" cross section in a single run of MCEEP, the event loop is executed twice. Thus if N events are requested, the program actually cycles through $2N$ events. The first set of N events give the radiative tail calculation and the second set gives the peak calculation where the kinematics are calculated assuming no radiation. In the second pass, the Schwinger correction is applied to the cross sections. The results of the Schwinger-corrected peak and the radiative tail are summed, so that all histograms and statistics reflect the full calculation.

4.1.1 Schwinger Correction

The schwinger correction is taken from Penner [6] and has the form

$$C_{schwing} = e^{-\delta_r}\big(1 - \delta_v\big) \tag{25}$$

where δ_r is the soft photon contribution and δ_v is the virtual photon contribution. These factors are given by

$$\begin{aligned}\delta_r &= \frac{\alpha}{\pi}\Big(\ln\frac{Q^2}{m_e^2} - 1\Big)\ln\Big[\frac{e^2}{\xi^3}\frac{1}{\Delta^2}\Big] \\ \delta_v &= \frac{\alpha}{\pi}\Big[\frac{28}{9} - \frac{13}{6}\ln\frac{Q^2}{m_e^2} + \frac{\pi^2}{6} - L_2(\cos^2\frac{\theta_e}{2})\Big]\end{aligned} \tag{26}$$

for elastic scattering, and by

$$\begin{aligned}\delta_r &= \frac{\alpha}{\pi}\Big(\ln\frac{Q^2}{m_e^2} - 1\Big)\ln\Big[\frac{\kappa}{\xi^2}\frac{ee'}{\Delta^2}\Big] \\ \delta_v &= \frac{\alpha}{\pi}\Big[\frac{28}{9} - \frac{13}{6}\ln\frac{Q^2}{m_e^2} + \frac{1}{2}\ln^2\frac{e}{e'} + \frac{\pi^2}{6} - L_2(\cos^2\frac{\theta_e}{2})\Big]\end{aligned} \tag{27}$$

for inelastic scattering, where

$$\begin{aligned}\xi &= 1 + \frac{e}{M_T}(1 - \cos\theta_e) \\ \kappa &= 1 + \frac{\omega}{M_T}(1 - \cos\theta_e).\end{aligned} \tag{28}$$

$L_2(x)$ is the Spence function defined by

$$L_2(x) = -\int_0^x \frac{\ln(1-y)}{y}dy \tag{29}$$

and Δ is the photon cutoff energy.

The Schwinger correction is applied to the "peak" cross section as an overall multiplicative factor. The kinematics are evaluated assuming no radiation, which is sensible provided the cutoff energy is not too large.

4.1.2 Radiative Tail

Here the kinematics of the scattering event are affected by the radiation of a hard photon. In general, the photon can be emitted in any direction. However, for the present implementation the peaking approximation is used, where the photon is assumed to be radiated either along the incident or scattered electron directions. This choice also keeps the electron on its mass shell. In order to not be locked into this choice, the kinematic routines no longer assume that the electron is on the mass shell. Thus, the energies **and** 3-momenta are now both passed to KINEM.

Calculation of the radiative tail requires sampling the photon momentum as well. In peaking approximation, this amounts to sampling the photon energy and then deciding whether the photon is emitted along the incident or scattered electron directions. This choice is made randomly (*i.e.* a "coin toss"). If both possibilities lead to acceptable events, a weighting factor of two is used to account for the fact that each is selected only half the time. Given the structure of MCEEP, this method is simpler than calculating both cases per event, since each case would lead to a different kinematics.

In order to keep changes to a minimum the following sampling schemes are used.

4.1.2.1 Elastic Radiative Tail

For elastic scattering, the electron angles (ϕ_e and θ_e) are randomly sampled. In addition, the photon energy, k, is sampled according to a $1/k$ distribution. Thus, the energy is chosen according to:

$$k = \Delta\Big(\frac{k_{max}}{\Delta}\Big)^x \tag{30}$$

where x is uniformly sampled between 0 and 1 and k_{max} is the upper limit for the photon energy (chosen to be the low edge of the scattered electron momentum acceptance). This implies a normalization factor of $\ln(k_{max}/\Delta)$ for the integral. In addition, the radiated cross section must then be divided by $1/k$ to account for the non-uniform weighting.

Once the electron angles and photon energy are chosen, the remaining kinematical quantities are calculable for the two cases of radiation before and after the primary scattering. Of course, this is only true in peaking approximation where the photon angles are then fixed for each of the two cases. Defining a so-called "pseudo-beam" 4-vector as

$$\epsilon_\mu = (\epsilon_0, \vec{\epsilon}) = e_\mu - k_\mu \tag{31}$$

where e_μ and k_μ are the incident electron and photon 4-momenta respectively, we have for the asymptotic energy of the scattered electron:

$$e' = \frac{M_T\,\epsilon_0 + \epsilon_\mu^2/2}{\epsilon_0 - |\vec{\epsilon}|\cos\alpha + M_T} \tag{32}$$

where α is the angle between $\vec{\epsilon}$ and $\vec{e}\,'$. Here, terms of order m_e^2 have been neglected. Note that this kinematic relation is general and does not assume peaking approximation. (For no radiation, the above relation reduces to the usual relation for elastic scattering: $e' = e/(1 + 2e/M_T \sin^2(\theta_e/2))$.) Treating ϵ_μ as an effective beam 4-vector allows calculation of the remaining kinematical quantities in the same way as for the case of no radiation. For example the 4-momentum transfer is just $\epsilon_\mu - e'_\mu$ and this is also the 4-momentum of the recoiling nucleus.

The prescription of Mo and Tsai [7] was used to calculate the elastic radiative tail in peaking approximation (see Equation (C.11) from that reference). In terms of my notation:

$$\frac{d^3\sigma}{dk d\Omega_e} = \left|\frac{\partial e'}{\partial k}\right| \left[\frac{t_s}{k}\,\frac{M_T + (e-k)(1-\cos\theta_e)}{M_T - e'(1-\cos\theta_e)}\,\frac{d^2\sigma(e-k)}{d\Omega_e} + \frac{t_p}{k}\,\frac{d^2\sigma(e)}{d\Omega_e}\right] \tag{33}$$

where the first term corresponds to pre-radiation and the second to post-radiation (in the MCEEP implementation only one of the two terms (either pre- or post-radiation) is calculated per event) and where

$$\begin{aligned} t_{s,p} &= \frac{\alpha}{\pi}\Big\{\frac{1}{2}(1+x_{s,p}^2)\ln\Big[\frac{2(sp)}{m_e^2}\Big] - x_{s,p}\Big\} \\ x_s &= \frac{e-k}{e} \\ x_p &= \frac{e'}{e'+k} \\ sp &= e_\mu e'_\mu = ee'(1-\cos\theta_e). \end{aligned} \tag{34}$$

The Jacobian, $\partial e'/\partial k$, accounts for the fact that MCEEP samples the photon energy, rather than the asymptotic scattered electron energy. This Jacobian has been calculated for the general case (*i.e.* not assuming peaking approximation) with care taken to give a stable result as the photon energy approaches zero (*i.e.* no division by k is involved). It is given by:

$$\frac{\partial e'}{\partial k} = \frac{e'}{M_T\,\epsilon_0 + \epsilon_\mu^2/2}\Big\{-M_T - (e-e') + (\vec{e}-\vec{e}\,')\cdot\hat{k}\Big\} \tag{35}$$

where $\hat{k}$ is a unit vector in the direction of the radiated photon. In the limit of no radiation ($k \to 0$) we have:

$$\frac{\partial e'}{\partial k} \to -\frac{1}{M_T}\frac{e'}{e}\Big(\omega + M_T - \vec{q}\cdot\hat{k}\Big). \tag{36}$$

4.1.2.2 Bound State Radiative Tail

For the bound state problem, the asymptotic scattered electron momentum (corresponding to the value of the momentum actually observed in an experiment) and hadron angles are sampled as well. This case is simpler than the elastic case, since the scattered electron energy is sampled and therefore known. Thus, the 4-momentum transfer can easily be calculated (taking into account the radiated photon as well) by:

$$q_\mu = \epsilon_\mu - e'_\mu = e_\mu - e'_\mu - k_\mu. \tag{37}$$

The outgoing hadron momentum can now be calculated in the same way as for no radiation, since the calculation only depends on q_μ and various hadronic variables. As for the no-radiation case, we must also keep track of the number of acceptable solutions (zero, one or two) for the hadron momentum. If there are two solutions, then this event must carry an additional weighting factor of two, since only one solution is actually selected. Note that the number of solutions (and therefore the weight) can depend on whether pre- or post-radiation was selected for the event.

The prescription of Borie and Drechsel [8] is used for the calculation of the radiative tail in peaking approximation (this prescription is also used for the continuum problem).

This gives the $n+1$-fold differential cross section including photon emission in terms of the n-fold differential radiation-free cross sections. For the bound-state case:

$$\frac{d^6\sigma}{dk\,d\omega\,d\Omega_e\,d\Omega_p} = \frac{\alpha}{\pi k}\left[\frac{(e'+k)^2+e'^2}{(e'+k)^2}\frac{d^5\sigma(e,e'+k)}{d\omega\,d\Omega_e\,d\Omega_p}\ln\frac{2e'}{m_e} + \frac{e^2+(e-k)^2}{e^2}\frac{d^5\sigma(e-k,e')}{d\omega\,d\Omega_e\,d\Omega_p}\ln\frac{2e}{m_e}\right] \tag{38}$$

where the first (second) term corresponds to radiation after (before) the primary scattering. In the MCEEP implementation only one of the two terms (either pre- or post-radiation) is calculated per event.

4.1.2.3 Continuum Radiative Tail

For the continuum problem, the hadron momentum is sampled as well. The various derived kinematical quantities are then readily obtained from the sampled quantities (including the photon momentum).

Here, the Borie and Drechsel prescription is again used to calculate the radiative tail. The formula used is the same as for the bound state case except now all the cross sections are also differential in the missing mass, ϵ_m.

4.1.2.4 Multi-Photon Correction

This correction is optionally applied to the radiative tail (depending on the user's choice). A flux conservation argument is used to derive an approximate form for the multi-photon contribution to the radiative tail (see reference [9]). The MCEEP multi-photon factor also includes the virtual photon part of the Schwinger correction. The full correction is given by

$$f_{mp} = (1-\delta_v)\frac{(1-e^{-\delta_r})}{\delta_r}$$

where δ_r and δ_v are the two parts of the Schwinger correction (given above). This factor is applied to the tail cross section, event by event. The kinematics used to compute δ_r and δ_v are the vertex kinematics and not the asymptotic ones. As these terms only vary slowly with kinematics, the results should normally not be too sensitive to this choice.

4.2 External Processes

[This section was modified from text contributed by Luminita Todor.]

4.2.1 Hadron Energy Loss by Collisions

The average energy loss is calculated from the well-known Bethe-Bloch formula [10]:

$$-\frac{dE}{dx} = 2\pi N_a r_e^2 m_e c^2 \rho \frac{Z}{A}\frac{z^2}{\beta^2}\left[\ln\Big(\frac{2m_e\gamma^2 v^2 W_{max}}{I^2}\Big) - 2\beta^2 - \delta - 2\frac{C}{Z}\right] \tag{39}$$

with

dE/dx:	mean energy loss per unit path length
r_e:	classical electron radius (2.818 fm)
$m_e c^2$:	electron rest energy (0.511 MeV)
N_a:	Avogadro's number (6.022×10^{23} mol^{-1})
I:	mean excitation potential of material
Z:	atomic number of absorbing material (target)
A:	atomic weight of absorbing material
ρ:	density of absorbing material
z:	charge of incident particle in units of e
β:	v/c of the incident particle
γ:	$1/\sqrt{1-\beta^2}$
δ:	density correction
C:	shell correction
W_{max}:	maximum energy transfer in a single collision

The overall multiplicative constant is

$$2\pi N_a r_e^2 m_e c^2 = 0.1535 \text{ MeV} \cdot \text{cm}^2/\text{g}. \tag{40}$$

The maximum energy transfer is produced by a head-on or knock-on collision. For an incident particle of mass $M \gg m_e$ we can use the approximation

$$W_{max} = 2m_e c^2 \beta^2 \gamma^2. \tag{41}$$

For materials known to be used in JLAB experiments, the mean excitation potential of the target material was taken from tabulated values. However, for other materials, an empirical formula [10] was used to give a good approximation of this quantity.

The density and shell corrections to the Bethe-Bloch formula are important at high and low energies respectively. The density effect arises from the fact that the electric field of the particle tends to polarize atoms along its path. Because of this polarization, electrons far from the path of the particle will be shielded from full electric field intensity. Collisions with these outer lying electrons will therefore contribute less to the total energy loss than predicted by the Bethe-Bloch formula. Values for δ are given by a formula due to Sternheimer [11]:

$$\delta(X) = \begin{cases} \delta_0 10^{2(X-X_0)} & X < X_0 \\ 4.6052X + C_0 + a(X_1 - X)^m & X_0 < X < X_1 \\ 4.6052X + C_0 & X_1 < X \end{cases} \tag{42}$$

where $X = \log_{10}(\beta\gamma)$. The quantities X_0, X_1, C_0, a and m depend on the absorbing material. The parameter C_0 corrects for the difference between the material density at which these parameters were fitted (ρ_0) and the density at which they are used.

The shell correction accounts for effects which arise when the velocity of the incident particle is comparable to or smaller than the orbital velocity of the bound electrons. At such energies, the assumption that the electron is stationary with respect to the incident particle is no longer valid and the Bethe-Bloch formula breaks down. The correction is generally small and the following empirical formula[10] (valid for $\eta = \beta\gamma < 0.1$) is used to calculate it:

$$\begin{aligned} C(I,\eta) = &(0.422377\eta^{-2} + 0.034043\eta^{-4} - 0.00038106\eta^{-6}) \times 10^{-6} I^2 \\ &+ (3.850190\eta^{-2} - 0.1667989\eta^{-4} + 0.00157955\eta^{-6}) \times 10^{-9} I^3 \end{aligned} \tag{43}$$

where I is in eV. Since variations of the density and shell corrections within the experimental acceptance are typically very small, these corrections are evaluated only once per run of MCEEP (for the central kinematics).

4.2.2 Electron Energy Loss by Collisions

The basic mechanism of collision loss for electrons is same as for heavy charged particles. However, the Bethe-Bloch formula has a different form to take into consideration on one hand the fact that the maximum energy transfer $W_{max} = T_e/2$ (where T_e is the kinetic energy of the incident electron), and on the other hand the indistinguishability of the final electrons:

$$-\frac{dE}{dx} = 2\pi N_a r_e^2 m_e c^2 \rho \frac{Z}{A} \frac{1}{\beta^2} \left[\ln\left(\frac{\tau^2(\tau+2)}{2(I/m_e c^2)^2} \right) - F(\tau) - \delta - 2\frac{C}{Z} \right] \tag{44}$$

where τ is the kinetic energy of the electron in units of $m_e c^2$, and

$$F(\tau) = 1 - \beta^2 + \frac{\tau^2/8 - (2\tau+1)\ln 2}{(\tau+1)^2}. \tag{45}$$

4.2.3 Energy Loss Distribution (Straggling)

The previous formulas represent the mean energy loss suffered by charged particles when passing through a thickness of matter. For any given particle, the amount of energy lost will not, in general, be equal to this mean value because of statistical fluctuations in the number of collisions and in the energy lost in each collision. An initially monoenergetic beam, after passing through a fixed thickness of material, will show a certain distribution in energy that, on average, is reduced by an amount given by the Bethe-Bloch formula.

Calculating these distributions requires different theoretical treatments for various regions in the mean energy loss. To choose among these theories, the parameter κ representing the ratio between the mean energy loss and the maximum energy transfer possible in a single collision is used:

$$\begin{cases} \kappa \leq 0.01 & \text{Landau distribution (thin absorber)} \\ 0.01 < \kappa \leq 10.0 & \text{Vavilov distribution} \\ 10.0 < \kappa & \text{Gaussian distribution} \end{cases} \tag{46}$$

4.2.4 Multiple Scattering through Small Angles

Most of the small angle deflection suffered by a charged particle traversing a medium is due to Coulomb scattering from nuclei. The Coulomb scattering distribution is well represented by the theory of Moliere, but for small deflection angles it is roughly a Gaussian. The Moliere scattering formulas are expressed in terms of two angles – the characteristic angle χ_c and the screening angle χ_α, defined as [12]:

$$\begin{aligned} \chi_c^2 &= 0.157\big[Z(Z+1)X/A\big]\big[z/(p\beta)\big]^2 \\ \chi_\alpha^2 &= 2.007\times10^{-5}Z^{2/3}\big[1+3.34(Zz\alpha/\beta)^2\big]/p^2 \end{aligned} \tag{47}$$

where p is the momentum in MeV/c, X is the path length in g/cm^2, Z and A are the atomic number and atomic weight of the scattering material, z is the charge of the projectile, $\beta = v/c$, and $\alpha \approx 1/137$. A good representation of the Moliere distribution is a Gaussian distribution with the width σ calculated in the following way:

$$\sigma = \frac{\chi_c^2}{1+F^2}\Big[\frac{1+v}{v}\ln(1+v) - 1\Big] \tag{48}$$

with

$$v = 0.5\frac{\chi_c}{\chi_\alpha}\frac{1}{(1-F)}$$

and where the parameter F represents the fraction of Moliere events included in the Gaussian approximation (*i.e.* the large angle tail is excluded). For F between 0.900 and 0.995, the width of the Moliere distribution is reproduced to better than 2% by this representation.

4.2.5 External Bremsstrahlung

In addition to ionization energy loss, fast particles lose energy by interaction with the Coulomb field of the nuclei of the traversed medium. If the charged particles are accelerated in the Coulomb field of the nucleus, a fraction of their kinetic energy will be emitted in the form of photons (bremsstrahlung). In estimating this process it is convenient to measure the thickness of the material in units of the radiation length X_0. This is the mean distance over which a high energy electron loses all but $1/e$ of its energy by bremsstrahlung. To calculate the parameter $b = 1/X_0$ we use the following formula[10]:

$$b = \frac{1}{X_0} = 4\alpha\, r_e \frac{N_A}{A}\Big[Z^2(L_{rad} - f(Z)) + Z L'_{rad}\Big] \tag{49}$$

For $A = 1$ g/mol, the quantity $4\alpha\, r_e N_A/A$ =(716.408 g/cm^2)$^{-1}$. The parameters L_{rad} and L'_{rad} are tabulated or can be calculated with empirical formulas:

$$\begin{aligned} L_{rad} &= \ln(184.15\, Z^{-1/3}) \\ L'_{rad} &= \ln(1194\, Z^{-2/3}). \end{aligned} \tag{50}$$

The function $f(Z)$ is an infinite sum but can be very well represented by:

$$f(Z) = a^2\Big(\frac{1}{1+a^2} + 0.20206 - 0.0369a^2 + 0.0083a^4 - 0.002a^6\Big) \tag{51}$$

where $a = \alpha Z$. The probability distribution for a particle of energy E to radiate an energy ΔE when traversing t radiation lengths of material is:

$$I_{ext}(E, \Delta E, t) = \frac{bt}{\Gamma(1+bt)}\Big(\frac{\Delta E}{E}\Big)^{bt}\Big\{\frac{1}{\Delta E}\Big[1 - \frac{\Delta E}{E} + \frac{3}{4}\Big(\frac{\Delta E}{E}\Big)^2\Big]\Big\}. \tag{52}$$

To generate this external radiation distribution an accept-reject technique is used (in the same manner as in AEEXB [13]). Consider the envelope function:

$$\epsilon(\Delta E) = 1.1\frac{bt}{\Gamma(1+bt)E^{bt}}(\Delta E)^{bt-1}. \tag{53}$$

To create a sampled distribution having $\epsilon(\Delta E)$ as its shape, two random numbers r_1 and r_2 are generated. We start with a trial value:

$$\Delta E_{trial} = E(r_1)^{1/bt} \tag{54}$$

and impose a test

$$r_2\, \epsilon(\Delta E_{trial}) < I_{ext}(E, \Delta E_{trial}, t). \tag{55}$$

Enforcing this test ensures that the generated ΔE_{trial} follows the desired bremsstrahlung distribution. If the test is failed for a given event, the calculation is restarted with a new set of random numbers r_1 and r_2.

Chapter 5

OUTPUT OF MCEEP

The various histogram/Ntuple options are now described, including the specification of cuts (both global and histogram specific) as well as the preprocessor routine RANGES which finds default limits for each histogram. Transport histograms/Ntuples are not discussed here but are described in the spectrometer analysis chapter. Also the method of integration over the experimental acceptance to produce distributions of various physics observables is described here.

5.1 Physics Observables Computed by MCEEP

Each histogram consists of some dynamical quantity *vs.* a kinematical variable both of which are selected by the user. The user specifies any desired histograms via the input file. For each histogram a "weighting" tag which specifies the dynamical variable and an index corresponding to a particular kinematical variable are specified. The user may also specify a set of Ntuples by listing the indices of all variables to be included. The Ntuple specification must also include the weighting tag.

In the case of an unpolarized target the coincidence cross section can be written as:[14]

$$\begin{aligned}\sigma &= \sigma^u \Big[1 + h\big(A + \vec{\pi}' \cdot \hat{S}\big) + \vec{\pi} \cdot \hat{S}\Big] \\ &= \sigma^u \Big[1 + hA + \vec{\mathcal{P}} \cdot \hat{S}\Big]\end{aligned} \tag{56}$$

where σ^u is the unpolarized cross section, h is the beam helicity (from -1 to $+1$), A is the electron analyzing power and $\hat{S}$ defines the direction of quantization of the hadron spin. The quantity $\vec{\pi}$ ($\vec{\pi}'$) is a vector consisting of the helicity independent (helicity dependent) partial cross sections for each spin orientation (n, t and l). Finally, $\vec{\mathcal{P}}$ is the sum of $\vec{\pi}$ and $h\vec{\pi}'$.

From the above expression, we define several "partial cross sections" (each of which is a linear combination of various response functions):

$$\begin{aligned}\sigma^A &= \sigma^u hA \\ \sigma^{\mathcal{P}_\alpha} &= \sigma^u \mathcal{P}_\alpha \qquad (\alpha = n, t, l) \\ \sigma^{\pi_\alpha} &= \sigma^u \pi_\alpha \qquad (\alpha = n, t, l) \\ \sigma^{\pi'_\alpha} &= \sigma^u h\pi'_\alpha \qquad (\alpha = n, t, l)\end{aligned} \tag{57}$$

where σ^A is the partial cross section associated with the electron analyzing power, $\mathcal{P}_\alpha$ refer to the three components of $\vec{\mathcal{P}}$, the ejectile polarization vector, π_α refer to the helicity independent components of $\vec{\mathcal{P}}$ and $h\pi'_\alpha$ the helicity dependent components.

MCEEP allows examination of σ^u, hA or any of the nine ejectile polarization components ($\mathcal{P}_\alpha$, π_α, $h\pi'_\alpha$ with $\alpha = n, t, l$) averaged over the experimental acceptance (see below for details) in addition to the corresponding partial yields versus various kinematic variables. (Also, various single-arm cross sections can be examined.) The options are summarized in Table 1. The weighting "tag" (see Table 1) can be positive or negative which will result in histograms/Ntuples of partial yields or cross sections respectively. For example, -1 would result in a histogram of the (e,e′X) cross section vs. some kinematical variable. The cross sections are computed by dividing the appropriate yield array by the phase space array, channel by channel. For Ntuples, this is performed by executing a "kumac" macro (see the file: *mc.kumac* in the utilities directory) after running MCEEP. (First the MCEEP Ntuple must be converted to HBOOK format by using *mceep_util* in the "utilities" directory.) Coincidence cross sections (Tag $= -1$) are in fm^2/(MeV-MeV/c-sr^2) for the continuum, fm^2/(MeV-sr^2) for bound states and fm^2/sr for elastic scattering. Note for the continuum case the cross sections are differential in the hadron momentum, not the energy. For the singles cross sections (Tag $= -100, -101, -102, -103$) the units are nb/(MeV/c-sr). The polarization histograms are obtained by dividing the yield histogram for a given spin orientation (either helicity dependent, independent or the sum of both) by the unpolarized cross section histogram, channel by channel. (A similar procedure for Ntuples can be invoked by using the file *mc.kumac*.)

The kinematical variable(s) for a histogram or Ntuple is (are) selected by specifying indices within the array, VAR. The elements of this array are specified in the Fortran INCLUDE file *var.cmn*. The variables currently available within MCEEP along with their indices are given in Appendix A.

5.2 Integrating Over the Experimental Acceptance

MCEEP performs integrals of the partial cross sections to compute partial yields as well as cross sections or polarization components averaged over the experimental acceptance. By calculating integrals over the appropriate phase space volume, MCEEP can simulate:

i.) Unconstrained coincidence (e,e′X) reactions (*i.e.* scattering to the continuum of the residual system);

ii.) Coincidence (e,e′X) reactions leading to a bound state of the residual nucleus;

iii.) Elastic scattering with detection of the recoil nucleus (*i.e.* A(e,e′A)). In addition, for elastic scattering, the user may elect to not enforce the hadron arm acceptance cuts (*i.e.* single-arm elastic scattering).

In general, for an (e,e′X) coincidence measurement with a beam of fixed energy and orientation, there are six variables needed to describe the reaction in the laboratory. These six variables specify the four-momenta of the scattered electron and ejectile (for a given ejectile mass). For scattering to discrete states of the residual nuclear system, in which case the missing mass (*i.e.* nuclear excitation) is constrained, one effectively has a five-fold acceptance volume. Finally, for elastic scattering from a nucleus where the reaction kinematics are completely determined from the scattering angle, a two-fold acceptance volume results (corresponding to the two orientation angles of the scattered

Tag	Weight
0	phase space (no cross section weighting)
±1	(e,e$'$X)
±2	(e,e$'$X) for N-type polarization (R frame) (total)
±3	(e,e$'$X) for T-type polarization (R frame) (total)
±4	(e,e$'$X) for L-type polarization (R frame) (total)
±5	(e,e$'$X) for N-type polarization (S frame) (total)
±6	(e,e$'$X) for T-type polarization (S frame) (total)
±7	(e,e$'$X) for L-type polarization (S frame) (total)
±8	(e,e$'$X) for X polarization in scatt plane (total)
±9	(e,e$'$X) for Y polarization in scatt plane (total)
±10	(e,e$'$X) for Z polarization in scatt plane (total)
±11	(e,e$'$X) $h\times$ electron analyzing power
±12	(e,e$'$X) for N-type polarization (R frame) (hel. indep.)
±13	(e,e$'$X) for T-type polarization (R frame) (hel. indep.)
±14	(e,e$'$X) for L-type polarization (R frame) (hel. indep.)
±15	(e,e$'$X) for N-type polarization (S frame) (hel. indep.)
±16	(e,e$'$X) for T-type polarization (S frame) (hel. indep.)
±17	(e,e$'$X) for L-type polarization (S frame) (hel. indep.)
±18	(e,e$'$X) for X polarization in scatt plane (hel. indep.)
±19	(e,e$'$X) for Y polarization in scatt plane (hel. indep.)
±20	(e,e$'$X) for Z polarization in scatt plane (hel. indep.)
±22	(e,e$'$X) for N-type polarization (R frame) (hel. dep.)
±23	(e,e$'$X) for T-type polarization (R frame) (hel. dep.)
±24	(e,e$'$X) for L-type polarization (R frame) (hel. dep.)
±25	(e,e$'$X) for N-type polarization (S frame) (hel. dep.)
±26	(e,e$'$X) for T-type polarization (S frame) (hel. dep.)
±27	(e,e$'$X) for L-type polarization (S frame) (hel. dep.)
±28	(e,e$'$X) for X polarization in scatt plane (hel. dep.)
±29	(e,e$'$X) for Y polarization in scatt plane (hel. dep.)
±30	(e,e$'$X) for Z polarization in scatt plane (hel. dep.)
±50	(e,e$'$X) accidentals for e',p or e',n (for (e,e$'$p) or (e,e$'$n) respectively)
±51	(e,e$'$X) accidentals for e',π^+ or e',π^0 (for (e,e$'$p) or (e,e$'$n) respectively)
±52	(e,e$'$X) accidentals for π^-,p or π^-,n (for (e,e$'$p) or (e,e$'$n) respectively)
±53	(e,e$'$X) accidentals for π^-,π^+ or π^-,π^0 (for (e,e$'$p) or (e,e$'$n) respectively)
±100	(e,e$'$)
±101	(e,π^-)
±102	(e,p) or (e,n) (for (e,e$'$p) or (e,e$'$n) respectively)
±103	(e,π^+) or (e,π^0) (for (e,e$'$p) or (e,e$'$n) respectively)
±121–130	(e,e$'$p) for multiple calculations per run

Table 1 *Histogram/Ntuple Weighting Options for MCEEP. Positive tags correspond to yields or partial yields. Negative tags correspond to average cross sections or polarizations. See the text for units of the cross sections.*

electron in the laboratory). For convenience, we introduce the following shorthand notation:

$$\mathcal{S}_i[\mathcal{O}] \equiv \begin{cases} \int_{\Delta v_i} d\omega dp d\Omega_e d\Omega_p \mathcal{O}, & \text{continuum;} \\ \int_{\Delta v_i} d\omega d\Omega_e d\Omega_p \mathcal{O}, & \text{bound state;} \\ \int_{\Delta v_i} d\Omega_e \mathcal{O}, & \text{elastic.} \end{cases} \tag{58}$$

This gives the integral of an observable, $\mathcal{O}$, over the multi-dimensional phase space consistent with the experimental acceptance volume, Δv_i. The subscript i indicates that the integral should extend only over the volume of a given kinematical bin (the i dependence thus describes the kinematical dependence of the integral). In the case of an unconstrained coincidence reaction (*i.e.* continuum), it is assumed that the observable $\mathcal{O}$ (which refers to one of the partial cross sections defined above) is differential in the electron energy loss ($d\omega$), the hadron momentum (dp), and the electron and hadron solid angles ($d\Omega_e$ and $d\Omega_p$ respectively). MCEEP obtains the unpolarized yield from

$$y_i^u = \mathcal{L}\Delta t \mathcal{S}_i[\sigma^u] \tag{59}$$

where $\mathcal{L}$ is the average luminosity and Δt is the counting time.

The same procedure is used to find other partial yields from each of the partial cross sections defined earlier. Further, acceptance averaged values of various physics observables are obtained by taking ratios of the appropriate partial yields. We have

$$\begin{aligned} < \sigma^u >_i &= \mathcal{S}_i[\sigma^u]/\mathcal{S}_i[1] \\ < hA >_i &= \mathcal{S}_i[\sigma^A]/\mathcal{S}_i[\sigma^u] \\ < \mathcal{P}_\alpha >_i &= \mathcal{S}_i[\sigma^{\mathcal{P}_\alpha}]/\mathcal{S}_i[\sigma^u] \\ < \pi_\alpha >_i &= \mathcal{S}_i[\sigma^{\pi_\alpha}]/\mathcal{S}_i[\sigma^u] \\ < h\pi'_\alpha >_i &= \mathcal{S}_i[\sigma^{\pi'_\alpha}]/\mathcal{S}_i[\sigma^u] \end{aligned} \tag{60}$$

for the unpolarized cross section, electron analyzing power and hadron polarization components respectively, where $\mathcal{S}_i[1]$ is the phase space volume for kinematic bin i. Note that the above method results in acceptance averaged observables weighted by the unpolarized cross section.

5.3 Histogram Range Determination

Since, in many cases, the user does not know ahead of time what the histogram variable ranges are, MCEEP provides defaults. The user may either specify the ranges of a given histogram within the input file or, by leaving these values blank, have the program compute them. MCEEP accomplishes this by calling the preprocessor subroutine, RANGES. RANGES uses a uniform grid over the experimental acceptance to determine the minimum and maximum values sampled for all kinematical variables. The routine which calculates these variables, KINEM, is called both in RANGES and in the main program event loop. This makes the addition of new kinematical variables simple, since the user need only add the new variable to one subroutine, KINEM.

To enhance program execution speed, RANGES may be run with a coarse grid (the default grid employs 6 iterations (from 0 to 5) in each of the six variables). However, with a coarse grid the routine will occasionally find ranges which are significantly smaller than the actual limits. To partially compensate for this, a 10% margin is added at each end of the range. However, this may still produce a range which is too small and this will be evidenced by a large number of out-of-range events in the corresponding histograms. In this case, MCEEP can be run again with limits explicitly entered by the user or with a finer grid. In other cases RANGES may find no or only one physical solution within its coarse grid, in which case the range cannot be determined. If this happens the routine will prompt the user to type in values for the histogram limits. This obviously only makes sense if the program is run interactively.

5.4 Histogram Cuts

Cuts can be placed on any histogram variable. All cuts are specified within the input file. There are two options supported by the program: *global* and histogram *specific* cuts. Global cuts affect all histograms and Ntuples whereas histogram specific cuts affect only the histograms to which they are attached. Thus, for example, the same histogram may be examined with many different cuts in a single analysis. Note that global cuts affect even histograms with histogram specific cuts attached as well as the Transport histograms and Ntuples. For details regarding the format of the input, see the section on the input file.

5.5 Options for Producing Output

There are currently four options for outputting results, three of which produce files suitable for the Topdrawer plotting program. The selection is made by specifying a CHARACTER*3 name within the input file. For details regarding the format of the input, see the section on the input file.

′P1D′ This option produces a regular Topdrawer histogram.

′P2D′ This option produces a three-dimensional Topdrawer histogram.

′SCA′ This option produces a two-dimensional Topdrawer scatter plot.

′NTU′ This option produces a MCEEP Ntuple.

Note that since Topdrawer does not allow variable density with each point, the ′SCA′ option reflects only the phase space distributions and reveals nothing about the physics weighting.

It was intended that MCEEP be a self-contained program, to avoid the need for users to link to other libraries. Hence, HBOOK type output is not supported by MCEEP. However, the companion program MCEEP_TO_HBOOK can be used to produce HBOOK files from histogram or Ntuple files generated by MCEEP (see Appendix B).

Chapter 6

SPECTROMETER ANALYSIS

MCEEP provides a flexible method for incorporating spectrometer/detector properties and associated uncertainties into the analysis. For each event a target Transport vector is constructed (one vector for each arm). This vector is then modified within the spectrometer analysis subroutine, SPECTROMETER, which produces an “affected” target Transport vector. In addition histograms, Ntuples and cuts can be made at any point along the spectrometer line. Note that, as mentioned above, the so-called “spectrometer” elements can be used to examine various non-magnetic devices as well as spectrometers.

6.1 Analysis of Uncertainties

The spectrometer input section can be used to estimate the effects of various uncertainties or resolution smearing. This is accomplished by adding either fixed or gaussian distributed amounts to a given Transport ray coordinate. By comparing the results with and without this addition (in two separate runs), its effect can be evaluated. Further, these additions can be incorporated at two levels, selected by the user. The user may elect to affect the ray coordinate for the calculation of the cross section only so that the bin which the event populates does not change or he may have the addition affect only the bin but not the cross section itself. (Also, the user may elect to have both the bin and cross section affected.)

The first method of affecting only the cross section has a very important advantage. By not affecting the kinematics as far as which bins are sampled, one insures that all bins receive exactly the same statistical sample as they would if the ray coordinates were not modified. Thus, in this case, differences which result from the change in the ray coordinate reflect only physics variations and have no component due to a changing statistical sample. Using this method allows the estimation of very small uncertainties without necessitating a commensurately large number of events. In contrast, the second method would cause events to shift from bin to bin due to offsets or resolution smearing. The second method would therefore require a very large number of events so that statistical errors do not completely dominate the observed differences.

There are cases, however, where the second method of applying the offsets or resolution smearing at the level of the kinematical bins is preferable. For constrained kinematics (elastic scattering or (e,e$'$X) to bound states) changing one variable implies changes in other correlated variables. The cross section should not be evaluated without modifying these correlated variables as well since this would be clearly unphysical. Therefore, for these cases, it may be preferable to change the kinematics at the level of the bins rather than cross sections. In this case, the event is seen to have “pristine” kinematics from which the physics is determined. Uncertainties only affect which bin is actually populated.

6.2 Options

The following spectrometer analysis options are currently supported and can be selected in the input file:

'HRS' This option is specific to the JLAB Hall A high resolution spectrometers (HRS). The Transport vector at the focal plane is produced assuming that the current (*i.e.* input) vector corresponds to the target. In addition, separate maps from the target to each limiting aperture are performed and the user may elect to enable or disable each of the corresponding tests. The global cut flag is set accordingly. Thus, events which do not pass all the enabled aperture tests result in a failed global cut.

'HRI' This option is specific to the JLAB Hall A high resolution spectrometers (HRS). The Transport vector at the target is produced assuming that the current (*i.e.* input) vector corresponds to the focal plane. Thus, this corresponds to the HRS Inverse map. The apertures are NOT tested for the inverse map. This option, following the HRS option, allows incorporating measurement uncertainties and/or multiple scattering at the focal plane, since the resulting target coordinates will then be affected.

'MAT' This results in multiplication of the current Transport vector by a MATrix. The matrix must be supplied by the user and must reside in the directory mceep/dat. The resulting vector is then the starting point for subsequent operations or histograms or cuts. This option allows for a crude representation of the spectrometer map by either a first-order or second-order Transport matrix. Future refinements may include a more general functional form for the spectrometer transformations.

'DFT' This has the effect of performing a DriFT. It is assumed that the trajectories are straight lines and hence this option only applies to field-free regions. The exact transformations of the ray coordinates are used (rather than a first order matrix).

'ROT' This performs a ROTation of the Transport vector. It is assumed that the trajectories are straight lines and hence this option only applies to field-free regions. The exact transformations of the ray coordinates are used. Note that this involves tracing rays to the new (*i.e.* rotated) coordinate system.

'TRK' This performs TRacK reconstruction assuming two wire chambers where each chamber consists of a pair of orthogonal wire planes. Offsets and resolution smearing can be included. The resolution effects include both multiple scattering and measurement errors. In the first case, the particle trajectory is affected for subsequent elements or wire planes. In the second case, the determination of the coordinates in the chambers is affected, but the particle trajectory is undisturbed.

'TOF' This provides the program with reference points in each spectrometer allowing a calculation of relative Time-Of-Flight.

'POL' Specifying this option directs the program to perform precession of the hadron POLarization through a magnetic element. The spin precession is approximated by assuming that the precession angle is a function of the net bend angle only. The bend angle is computed via a Transport matrix, so this option must be immediately *followed* by a 'MAT' element. Precession through a series of elements can be calculated by simply stacking POL and MAT elements in pairs.

'COS' Specifying this option directs the program to perform precession of the hadron polarization through a magnetic spectrometer using a COSY spin transfer matrix. A filename containing the COSY matrix must also be specified or else the word 'internal'. In the latter case, MCEEP will use a set of standard COSY outputs for the JLAB-Hall A spectrometers contained in the mceep/dat directory. These outputs have been generated for a set of proton momentum values; MCEEP then interpolates on the two outputs neighboring the spectrometer central momentum setting for the given input. These "standard" COSY files are specifically for proton ejectiles. Further, they assume that the particle coordinates refer to the values at the target. Thus, as the COSY matrix operates on the polarization vector only, it is necessary to include any transport of the momentum vector *AFTER* the 'COS' element.

'RES' This option results in convoluting a specified Transport coordinate with a gaussian distributed RESolution function. This allows effects of resolution smearing (*e.g.* multiple scattering) to be simulated. This operation can be applied to either the cross sections, kinematic bins or both. By default this operation is applied to the kinematic bins only since the multiple scattering occurs after the primary scattering event and hence cannot affect the cross section. (See the section on error analysis.)

'OFF' This adds a constant OFFset (specified by the user) to a particular Transport ray coordinate. This allows evaluation of the effects of misalignments (in position sensitive detectors, for example). This operation can be applied to either the cross sections, kinematic bins or both. By default this operation is applied to the cross sections only and not the kinematic bins. This gives a more accurate representation of an actual measurement since misalignments are presumably unknown. Thus, the bin into which the event is placed should not be affected. Rather the bin is illuminated by a cross section which reflects the misalignments. (See the section on error analysis.)

'H1D' This specifies that a histogram is to be made at the current position in the spectrometer line. Any one of the six Transport coordinates can be histogrammed. The histogram is affected by all operations which precede this element in the input file.

'H2D' This specifies that a three-dimensional histogram is to be made at the current position in the spectrometer line. Any one of the six Transport coordinates can be histogrammed versus any other. The histogram is affected by all operations which precede this element in the input file.

$'SCT'$ This specifies that a scatter plot is to be made at the current position in the spectrometer line. The histogram X and Y values are written to a file for each event. Every point in the histogram carries the same density. Thus, the resulting scatter plot does not represent the actual event population but merely reflects the phase space weighting.

$'NTU'$ This specifies that an Ntuple is to be made at the current position in the spectrometer line. The entire Transport vector is written to a file for each event, along with the appropriate weighting factor. The Ntuple is affected by all operations which precede this element in the input file.

$'CUT'$ This indicates that a cut is to be placed at the current position in the spectrometer line. There are essentially four options for the cuts (see the section on the input file specification for the correct format for each).

- $'G'$ or $'S'$ Selects <u>G</u>lobal or <u>S</u>pecific cuts. Global cuts affect all histograms and scatter plots, whether they precede or follow the current element. In addition, they affect the ordinary kinematics histograms. Specific cuts affect only those Transport histograms or scatter plots to which they are explicitly attached. They cannot be attached to the kinematics histograms.

- $'R'$ or $'E'$ Selects <u>R</u>ectangular or <u>E</u>lliptical cuts. Rectangular cuts refer to a single Transport coordinate. Rectangular apertures can be constructed by specifying two successive cuts on the appropriate coordinates. Elliptical cuts refer to a pair of Transport coordinates. In this case the limits for the cut refer to the semi-major/minor axes of the ellipse. For circular apertures these should be set equal to each other.

Any number and combination of the above elements can be stacked. Thus, for example one can apply a drift matrix to the scattering chamber exit window followed by multiple scattering in the exit window followed by a matrix from the exit window to the focal plane, followed by a detector misalignment, *etc.* The final action to be taken must give the target Transport vectors again, since it is from these objects that the physical variables, p_i, and dynamical quantities are computed. Thus, the above spectrometer elements should form a closed loop starting and ending at the target Transport vector, x_i^t and $x_i^{t'}$ respectively:

$$q_i \rightarrow x_i^t \rightarrow x_i^{t'} \rightarrow q_i' \rightarrow p_i. \tag{61}$$

This requirement can be avoided by choosing (within the MCEEP input file) to not apply the spectrometer elements to the kinematic quantities.

As indicated above, the $'$RES$'$ and $'$OFF$'$ options can affect either the binning variables, the cross sections or both according to the user's preference. (Defaults are provided which result in a simulation which accurately resembles the actual experiment (*i.e.* cross sections for $'$OFF$'$ and kinematical bins for $'$RES$'$).) The way in which MCEEP accomplishes this is by calling the SPECTROMETER subroutine twice. Immediately after the first call all the cross sections (both single arm and coincidence) are computed.

Only those elements which are to affect the cross sections are applied in the first call to SPECTROMETER. After the cross sections are computed, the Transport target vectors (one for each spectrometer) are reset to their original values. The SPECTROMETER subroutine is then called again and now all elements affecting the binning variables are applied. The variables for all requested Transport histograms/Ntuples as well as logical flags for each Transport histogram whose values depend upon whether all *specific* cuts are passed are obtained in the second call. (However, the histograms are actually incremented within the main routine, MCEEP, subsequent to the second call to SPECTROMETER.) Matrices ("MAT") and the JLAB-Hall A HRS optics ("HRS" and "HRI") are applied only in the second call, as well as drifts ("DFT"), rotations ("ROT"), spin precession ("POL" and "COS"), track reconstruction ("TRK") and the time-of-flight calculation ("TOF"). (Note that this is a change as of Version 3.1. Previously, all these elements, other than "HRS" and "HRI", were applied in both calls. This could lead to serious errors since the "TRK" calculation would be performed in the first call without the corresponding target to focal plane transport.) Although the method of calling SPECTROMETER twice is not the most efficient, it is simpler than performing only one call and keeping two sets of variables, one for the cross sections and one for the kinematic bins.

Chapter 7

RUNNING MCEEP

In order to run MCEEP a user input file must be set up. This along with the output generated by MCEEP is now described.

7.1 A Sample MCEEP Run

Most of the input required by MCEEP is specified via an input file (see the next section). However, MCEEP prompts the user at run-time with menus of various physics models, *etc.* Note that the number of questions asked by the program depends on the input. A sample run is shown in Figure 4. The user's input is shown underlined; all other information is displayed by the program.

The possible responses to the various questions asked by MCEEP will now be summarized. See the chapter on physics models (and references quoted therein) for details on the menu options.

INPUT FILE

The name of the input file. MCEEP assumes the file will be located in mceep/input. The file extension "*.inp*" is assumed. The structure of the input file is described in the next section.

ELASTIC, BOUND STATE OR CONTINUUM

For elastic scattering (*i.e.* A(e,e′A)), bound state scattering or scattering to the continuum, MCEEP will perform 2, 5 or 6 dimensional integrals respectively. If elastic scattering is chosen, MCEEP will prompt the user to select scattering from the proton, deuteron, triton, ^{3}He, ^{4}He or ^{12}C. If the deuteron is chosen the user must then select one of the four available models.[15] Also, for elastic scattering in general, the user has the option of enforcing the acceptance of the hadron arm or simulating single-arm elastic scattering where the hadron kinematics are ignored. If elastic scattering is chosen, MCEEP requests no further input. For the bound state or continuum options, MCEEP will prompt the user to choose from various models (see below and also the chapter on physics models).

```
mceep

Enter input file prefix (.inp) >v3_5
Elastic (E), Bound State (B) or Continuum (C)? b
Radiate?  (0=No, 1=Yes, 2=Yes with multiphoton corr.)  2
Cutoff energy in MeV (RET for 1 MeV default) >

S(p,E) X Sigma_CC1 (unpolarized) --------------- 100
S(p,E) X Sigma_CC1 (polarized - Van Orden) ----- 200
Gross/Dmitrasinovic Deuterium IA --------------- 300
p(e,e'p)pi0 ------------------------------------ 500
p(e,e'pi+)n ------------------------------------ 600
p(e,e'K+)Y ------------------------------------- 700
d(e,e'p)n Arenhoevel unpolarized responses ----- 804

Enter Option>200
(e,e'p) or (e,e'n)? (P/N)>p

Deuteron:   Krautschneider ------------------------- 10
            Saha parametric fit to data ------------ 11
            from EPC code -------------------------- 12
            Bernheim data -------------------------- 13
            Van Orden parameterization ------------- 14
            Paris ---------------------------------- 15
            Bonn ----------------------------------- 16
            Argonne V18 ---------------------------- 17
            CD Bonn -------------------------------- 18
            Gross ---------------------------------- 19

Enter Option or <RET> for more options>10
Compute singles and accidentals?  (Y/N)>n
--------------------------------
Doing Radiation Tail Calculation
--------------------------------
Event number:  1000
Event number:  2000
            ⋮
Event number:  50000
----------------------
Doing Peak Calculation
----------------------
Event number:  1000
Event number:  2000
            ⋮
Event number:  50000
FORTRAN STOP
```

Figure 4 *A sample MCEEP run. The user's input is shown underlined. Note that the number of questions asked by the program depends on the input.*

RADIATIVE CALCULATION

MCEEP now provides for calculation of radiative effects. Answering "0" to this question will turn off calculation of the internal radiative effects. Answering "1" will turn on these effects but without the multi-photon correction[9]. Answering "2" will also turn on the multi-photon correction. (External radiation can be turned on or off according to a parameter in the user input file.) If internal radiation is turned on the user must supply the photon cutoff energy (hitting a carriage return will select the default value of 1 MeV).

PHYSICS OPTION

MCEEP now supports 5 physics options, numbered 100, 200, 300, 500 and 600.

100 Factorized (e,e′N) calculation in PWIA. This option produces the four unpolarized response functions. If this option is selected, MCEEP will then display menus of spectral functions for various nuclei (see below).

200 Factorized (e,e′N) calculation in PWIA. This option produces polarization response functions as well. If no polarizations are measured and if the beam is unpolarized, this option will produce the same results as option 100. In this case option 100 is preferred since it requires less computation time. If this option is selected, MCEEP will then display menus of spectral functions for various nuclei (see below).

300 Relativistic (unfactorized) PWIA calculation for d(e,e′p)n. This option produces polarization observables.

500 Pion electroproduction in p(e,e′p)π^0. This option produces polarization observables as well. If this option is selected, the user will be prompted for the name of a resonance option file. The file *res_standard.dat*, located in mceep/dat, is provided for use with this option.

600 Pion electroproduction in p(e,e′π^+)n. The pion survival probability is taken into account via a weighting factor for the cross section.

(e,e′p) or (e,e′n)

This defines which form factors are to be used for certain physics options. Note that the ejectile mass is set within the input file and this option will not override the value. Note that physics options 300 and 500 (see below) are explicitly for proton ejectiles.

SPECTRAL FUNCTION

If a factorized (e,e′N) option is chosen (option 100 or option 200) the user must then select from a menu of spectral distributions. The distribution chosen then multiplies the off-shell *ep* (or *en*) cross section. Additionally, the spectral function may be multiplied by an overall constant which is specified in the input file (via the variable SPEC_FAC). The default normalizations of each spectral function are listed in the chapter on physics models. For tabulated spectral functions, exponential interpolations (with a quadratic exponent) are performed to obtain the value for a given kinematics. For continuum scattering, a missing mass distribution function file will be requested by the program as well.

10–15 Deuteron momentum distributions.

20–24 ^{3}He momentum distributions for 2-body breakup channel. Includes a 2-dimensional spectral function, $S(p_r, \epsilon_m)$, to describe the continuum (*i.e.* 3-body breakup) channel.

30–34 ^{4}He momentum distributions for the t+p breakup channel.

40–42 ^{16}O PWIA momentum distributions for $p_{1/2}$, $p_{3/2}$ and $s_{1/2}$ shells from Van Orden.

43–45 ^{16}O RDWIA momentum distributions for $p_{1/2}$, $p_{3/2}$ and $s_{1/2}$ shells from Udias.

80–85 ^{208}Pb DWIA momentum distributions for $2s_{1/2}$, $2d_{3/2}$, $1h_{11/2}$, $2d_{5/2}$, $1g_{7/2}$ and $1g_{9/2}$ from Lapikas.

150–158 Harmonic oscillator momentum distributions for a variety of shells.

160–162 These options allow user-provided files to be used for spectral distributions.

SINGLES CALCULATION

MCEEP provides for the calculation of singles cross sections and yields, if the user elects to have the program calculate them. The cross sections are currently based on the routines of Lightbody and O'Connell.[16] Answering "Y" (or "y") to this question, will cause MCEEP to call the routines which calculate these cross sections and will result in a much slower execution speed.

7.2 Input file

The structure of the input file is now discussed. In addition, each input variable is described in detail. Finally, a sample input file is shown.

7.2.1 Structure

The structure of the input file follows; the variables are described in detail later. Each line below corresponds to a single line in the input file. In cases where a line exceeds the page width, it is continued on the next line with indentation. Blank lines, horizontal rules and phrases within brackets do not appear in the input file but are used here for clarity or informational purposes only.

[ITERATIONS]

N_EVENT
NITER1, NITER2, NITER3, NITER4, NITER5, NITER6

[MASSES]

EJECT_MASS, EJECT_CHG, EM_BOUND

[KINEMATICS]

E0, PH_B, TH_B, PF_E, PH_E, TH_E, PF_P, PH_P, TH_P

[ACCEPTANCES]

ACC_EP, ACC_EM, ACC_PP, ACC_PM
SA_SHAPE_E, SA_SHAPE_P,DPH_E, DTH_E, DPH_P, DTH_P

[RATE PARAMETERS]

ALUM, BEAMTIME, SPEC_FAC

[SINGLES PARAMETERS]

PFERMI, EPS, EPSD

[TARGET PARAMETERS]

ATARG, ZTARG, DENS_TARG, ITARG_MOD, ELOSS_MOD
[For ITARG_MOD=1:] PHI_TARG, NFOILS, *[For J=1,NFOILS:]* TARG_LO(J), TARG_HI(J)
[For ITARG_MOD=2 or 3:] TARG_LO, TARG_HI
DRIFTE_N, DRIFTP_N

[BEAM PARAMETERS]

POL_BEAM, BEAMV, BEAMD, DUTY_FAC, DEL_TOF
FWHM_X_B, FWHM_Y_B, FWHM_PH_B, FWHM_TH_B, FWHM_E0
DEL_X_B, DEL_Y_B, DEL_PH_B, DEL_TH_B, DEL_E0

RAST_SHAPE, RAST_AMP_X, RAST_AMP_Y

[SPECTROMETER ANALYSIS]

C_PID, APPLY_TO_LAB, NEL, THETA_BP, OBJECT_PT(3)

[For I=1 to NEL:]

[OP=]'HRS', HRS_ID, *[For J=1 to 5:]* APERTURE_TEST(J)

[or]

[OP=]'HRI', HRS_ID

[or]

[OP=]'MAT', ORDER, INVERT, TRNSPT_FIL

[or]

[OP=]'DFT', DRIFT_LENGTH

[or]

[OP=]'ROT', ROT_AXIS, ROT_ANGLE

[or]

[OP=]'POL', BEND_ANGLE

[or]

[OP=]'COS', COSY_SPIN_FIL

[or]

[OP=]'TRK', WC_FIL

[or]

[OP=]'TOF', CRAY_PATH

[or]

[OP=]'RES', OB, NCOORD, RESMOD, RESVAL

[or]

[OP=]'OFF', OB, NCOORD, OFF

[or]

[OP=]'H1D', IPID_TR, TR_MINX, TR_MAXX, NXCHAN_TR, NCUTS_TR,
[For J=1 to NCUTS_TR:] ICUT_IND_TR(I,J), NCOORD, TR_FILE

[or]

[OP=]'H2D', IPID_TR, TR_MINX, TR_MAXX, TR_MINY, TR_MAXY,
NXCHAN_TR, NYCHAN_TR, NCUTS_TR,
[For J=1 to NCUTS_TR:] ICUT_IND_TR(I,J),
NCOORDX, NCOORDY, TR_FILE

[or]

[OP=]'SCT', TR_MINX, TR_MAXX, TR_MINY, TR_MAXY, NCUTS_TR,
[For J=1 to NCUTS_TR:] ICUT_IND_TR(I,J),
NCOORDX, NCOORDY, TR_FILE

[or]

[OP=]'NTU', IPID_TR, BEND_ANGLE_TOT, TR_FILE

[or]

[OP=]'CUT', CUT_DOMAIN, CUT_TYPE,
[for $CUT\_DOMAIN = 'G'$ and $CUT\_TYPE = 'R'$:]
TR_CUT_MIN, TR_CUT_MAX, NCOORD
[for $CUT\_DOMAIN = 'G'$ and $CUT\_TYPE = 'E'$:]

TR_CUT_X, TR_CUT_Y, NCOORDX, NCOORDY

[for CUT_DOMAIN $= 'S'$ *and CUT_TYPE* $= 'R'$ *:]*

TR_CUT_MIN, TR_CUT_MAX, NCOORD, TR_CUT_IND

[for CUT_DOMAIN $= 'S'$ *and CUT_TYPE* $= 'E'$ *:]*

TR_CUT_X, TR_CUT_Y, NCOORDX, NCOORDY, TR_CUT_IND

C_PID, APPLY_TO_LAB, NEL, THETA_BP

[For I=1 to NEL:]

repeat above input for the other arm

[GLOBAL CUTS]

NCUTS

[For I=1 to NCUTS:]

ICUT_VAR, CUT_MIN, CUT_MAX

[SPECIFIC CUTS]

NCUTS_I

[For I=1 to NCUTS_I:]

ICUT_IND, ICUT_VAR_I, CUT_MIN_I, CUT_MAX_I

[KINEMATICS HISTOGRAMS]

NPLOTS

[For I=1 to NPLOTS:]

[PLOT_TYPE=]'P1D', IPID, X_MIN, X_MAX, NX_CHAN, X_SC, X_OFF, NCUTS_PL, *[For J=1 to NCUTS_PL:]* ICUT_IND_PL(I,J), I_VAR, PLOT_FIL

[or]

[PLOT_TYPE=]'P2D', IPID, X_MIN, X_MAX, Y_MIN, Y_MAX, NX_CHAN, NY_CHAN, X_SC, Y_SC, X_OFF, Y_OFF, NCUTS_PL, *[For J=1 to NCUTS_PL:]* ICUT_IND_PL(I,J), IX_VAR, IY_VAR, PLOT_FIL

[or]

[PLOT_TYPE=]'SCA', X_MIN, X_MAX, Y_MIN, Y_MAX, X_SC, Y_SC, X_OFF, Y_OFF, NCUTS_PL, *[For J=1 to NCUTS_PL:]* ICUT_IND_PL(I,J), IX_VAR, IY_VAR, PLOT_FIL

[or]

[PLOT_TYPE=]'NTU', IPID, NVAR_NTU, *[For J=1 to NVAR_NTU:]* I_VAR(I,J), PLOT_FIL

[or]

[PLOT_TYPE=]'NTM', NWT_NTM,

[For J=1 to NWT_NTM:] I_WT(I,J),

NVAR_NTU,

[For J=1 to NVAR_NTU:] I_VAR(I,J),

PLOT_FIL

[COMMENT LINES]

Up to 10 comment lines (read as CHARACTER*80)

7.2.2 Variable Descriptions

Each of the variables contained within the input file is now described. They are grouped according to their order of appearance in the file which corresponds with the order given above. There is some duplication since the same variables may appear in different options under the same general category (for example, histogram limits appear in all three display options, 'P1D', 'P2D' and 'SCA').

ITERATIONS

N_EVENT Number of iterations (*i.e.* tries) for the Monte Carlo calculation. For a bound state case only a fraction of these will actually lead to physical solutions within the experimental acceptance. In addition, for an elliptical aperture, the coordinates are selected to lie within the superscribed rectangle and then a test is made to guarantee that the event lies within the specified ellipse. Thus, in this case the fraction of "survivors" is approximately $\pi/4$.

NITER1, ..., NITER6 The number of iterations to be used by the preprocessor routine RANGES in each of the six grid variables. These numbers only have relevance for the default histogram limits computed by MCEEP. They do not affect the statistical accuracy of the program. Since the number of grid points is given by the product of these six numbers (actually the number of points sampled = $\prod_{i=1}^{6}(N_i + 1)$ since each index starts at zero) they should be made as small as possible to enhance execution speed. MCEEP uses default values of 5 for any of these numbers which are left blank or set equal to zero. NITER1 through NITER6 correspond to TH_P, PH_P, TH_E, PH_E, PF_E, PF_P respectively (see below for the definition of these quantities).

MASSES

EJECT_MASS The mass of the ejectile in MeV.

EJECT_CHG The charge of the ejectile in units of the electron charge (integer). This variable only has relevance for the multiple scattering calculation (which is only performed if the energy loss calculations are selected).

EM_BOUND The missing mass in MeV. If elastic scattering or the continuum case is selected this parameter is ignored. For p(e,e$'$p)π^0, for example, this parameter should be set to the mass of the π^0.

KINEMATICS

E0, PH_B, TH_B The energy, in-plane and out-of-plane angles of the incident electron (units are MeV and degrees).

PF_E, PH_E, TH_E The momentum, in-plane and out-of-plane angles of the scattered electron (units are MeV/c and degrees).

PF_P, PH_P, TH_P The momentum, in-plane and out-of-plane angles of the detected hadron (units are MeV/c and degrees).

ACCEPTANCES

ACC_EP, ACC_EM The momentum acceptance of the electron arm in %. The first number corresponds to the maximum momentum and the second the minimum.

ACC_PP, ACC_PM The momentum acceptance of the hadron arm (see above).

SA_SHAPE_E(P) These are CHARACTER*1 variables specifying the shape of the solid angle defining aperture for the electron and hadron arms respectively. For SA_SHAPE_E(P) = $'$R$'$ a rectangular aperture is assumed. The angular acceptances which follow then correspond to the full length of each side of the rectangle (in mr). For SA_SHAPE_E(P) = $'$E$'$ an elliptical aperture is assumed. The angular acceptances then correspond to the lengths of each side of a superscribed rectangle (*i.e.* twice the semi-major and semi-minor axes of the ellipse).

DPH_E, DTH_E The in-plane and out-of-plane angular acceptance of the electron arm in milliradians (see above).

DPH_P, DTH_P The in-plane and out-of-plane angular acceptance of the hadron arm in milliradians (see above).

RATE PARAMETERS

ALUM Luminosity in μA-g/cm^2. This number must take into account isotopic abundance as well as the orientation of the target relative to the beam (for solid targets).

BEAMTIME The total beamtime in hours. This will affect the overall magnitude of the yield histograms but has no effect on the Monte Carlo statistics. BEAMTIME only enters into the normalization in generating yields from the computed cross sections. The statistical accuracy of the Monte Carlo calculation is affected only by N_EVENT.

SPEC_FAC The spectroscopic factor for the (e,e$'$N) process. This number multiplies the (e,e$'$N) spectral function when one is chosen from the menu.

SINGLES PARAMETERS

PFERMI The Fermi momentum (relevant only for calculations of the singles cross sections).

EPS The nucleon separation energy in MeV (relevant only for calculations of the singles cross sections).

EPSD The Δ separation energy in MeV (relevant only for calculations of the singles cross sections).

TARGET PARAMETERS

ATARG The mass number of the target nucleus. For a compound target only the nucleus of interest is specified. The luminosity should reflect this as well.

ZTARG The atomic number of the target nucleus (relevant only for calculations of the singles cross sections or energy loss calculations).

DENS_TARG Density of the target material in g/cm^3 (relevant only for energy loss calculations).

ITARG_MOD Target model (integer). This should be set to one for target foil(s), to two for the JLAB Hall A LH_2/LD_2 cryotarget and to three for the JLAB Hall A ^{3,4}He cryotarget.

ELOSS_MOD Integer variable specifying the level of the energy loss calculations. For ELOSS_MOD=0, no energy loss or multiple scattering is performed. For ELOSS_MOD=1, the energy loss and multiple scattering is taken into account. ELOSS_MOD=2 also accounts for energy loss and multiple scattering, but the mean energy loss (dE/dx) is removed. This allows comparison with data corrected for the mean energy loss.

PHI_TARG Horizontal angle of target foil(s) in degrees. It is assumed that all foils have the same angle. An angle of zero indicates that the foils are normal to the nominal beam. The sense of rotation is defined in the same way as for the spectrometers, so that a positive angle implies a rotation of the normal toward beam left.

NFOILS This is an integer specifying the number of target foils. Normally it would be set to one.

TARG_LO(J) The starting position in meters measured along the beam direction for the Jth foil. For the cryotarget (ITARG_MOD=2) this specifies the starting position of the cell.

TARG_HI(J) The ending position in meters measured along the beam direction for the Jth foil. For the cryotarget (ITARG_MOD=2) this specifies the ending position of the cell.

DRIFTE_N This is the nominal drift in meters to the solid angle defining aperture for the electron arm. "Nominal" refers to the distance between the nominal beam-target interaction point and the center of the aperture. This only has relevance for extended targets.

DRIFTP_N This is the nominal drift in meters to the solid angle defining aperture for the hadron arm. This only has relevance for extended targets.

BEAM PARAMETERS

POL_BEAM The polarization of the beam (from -1 to $+1$). For an unpolarized beam this parameter should be set to zero.

BEAMV For a dispersed beam this specifies the vertical extent of the beam at the nominal target position in $\pm$ mm. If the beam is bent so that it makes an angle θ_b with respect to the laboratory floor (*e.g.* by using a beam swinger to go out-of-plane) the vertical extent at the target will be BEAMV/$\cos\theta_b$. Division by $\cos\theta_b$ is done automatically by the program so that the input parameter, BEAMV, should not reflect the out-of-plane beam. For an achromatic beam this parameter should be set to zero.

BEAMD For a dispersed beam this gives the momentum spread in $\pm$%. The momentum and vertical extent of the beam are assumed to be related linearly. For an achromatic beam this parameter should be set to zero.

DUTY_FAC Accelerator duty factor in %. This is used to calculate accidentals and therefore only has relevance if the singles cross sections are calculated.

DEL_TOF Time-of-flight window in nsec. This is used to calculate accidentals and therefore only has relevance if the singles cross sections are calculated.

FWHM_X_B The full width at half maximum (FWHM) (in cm) of a gaussian distributed addition to the beam horizontal position.

FWHM_Y_B The FWHM (in cm) of a gaussian distributed addition to the beam vertical position.

FWHM_PH_B The FWHM (in mr) of a gaussian distributed addition to the beam in-plane angle. This can be used to estimate the effect of multiple scattering in the scattering chamber entrance window or target for example.

FWHM_TH_B The FWHM (in mr) of a gaussian distributed addition to the beam out-of-plane angle.

FWHM_E0 The FWHM (in % of E0) of a gaussian distributed addition to the beam energy. This can be used to estimate the effects of a finite spread in beam energy. For a monochromatic beam this value should be set to zero.

DEL_X_B An offset (in cm) to be added to the beam horizontal position.

DEL_Y_B An offset (in cm) to be added to the beam vertical position.

DEL_PH_B An offset (in mr) to be added to the beam in-plane angle.

DEL_TH_B An offset (in mr) to be added to the beam out-of-plane angle.

DEL_E0 An offset (in % of E0) to be added to the beam energy. This can be used to estimate the effect of inaccurate knowledge of the centroid beam energy.

RAST_SHAPE CHARACTER*1 variable specifying the shape of the beam raster. Set to ′R′ for a rectangular raster, ′E′ for an elliptical raster or ′N′ for no raster.

RAST_AMP_X The horizontal size of the raster (in meters). For a rectangular raster this is the peak-to-peak distance and for an elliptical raster this is the "diameter". For no raster, this value is ignored (though a value still must entered in the input file).

RAST_AMP_Y The vertical size of the raster (in meters). For a rectangular raster this is the peak-to-peak distance and for an elliptical raster this is the "diameter". For no raster, this value is ignored (though a value still must entered in the input file).

SPECTROMETER ANALYSIS

C_PID This is a CHARACTER*1 variable which identifies one of the two coincidence spectrometers. Thus, this variable can take two values, ′E′ or ′P′, depending on whether the electron arm or hadron (proton) arm spectrometer elements follow.

APPLY_TO_LAB This is a LOGICAL variable which determines whether the final Transport vector is used to compute a new set of laboratory coordinates. This allows examining focal plane coordinates (for example) without having to apply an inverse map to get back to the target. This variable should be set to "T" (true) in order to apply the final Transport vector to determine new laboratory coordinates and to "F" (false) otherwise.

NEL This specifies the number of entries associated with the specification of the spectrometer and associated offsets, *etc.* If one does not wish to incorporate the spectrometers into the analysis this number should be set to zero. Note, however, that the spectrometer bend-plane angle, THETA_BP (see below), must be included if one wishes to examine recoil polarization variables in the spectrometer frame.

THETA_BP If the spectrometer is to be incorporated the variable THETA_BP must be set properly. This variable defines the orientation of the spectrometer bend plane (in degrees). The following examples illustrate the convention used.

−90 Bends vertically upward

+90 Bends vertically downward

180 Bends leftward

0 Bends rightward

Here, left and right refer to the bending direction as one looks down the optic axis from the target.

OBJECT_PT(3) These are the three components (X, Y and Z respectively) of the spectrometer object point in the LAB system (in centimeters). Non-zero values correspond to a mispointing of the spectrometer since the object point is then not coincident with the origin of the LAB system.

OP A CHARACTER*3 variable which specifies the type of OPeration to be performed for a given transformation element. The input for each element depends on the type of operation specified. The possible options are listed below.

'HRS' JLAB Hall A high resolution spectrometer target to focal plane map including apertures.

'HRI' JLAB Hall A high resolution spectrometer focal plane to target (*i.e.* inverse) map.

'MAT' Multiplication of the current Transport vector by a matrix. The matrix must be supplied by the user and must reside in the mceep/dat directory.

'DFT' Perform a drift through a field-free region.

'ROT' Refer the Transport ray coordinates to a rotated system. This only works for field-free regions.

'POL' Precess the hadron spin through a magnetic element.

'COS' Precess the hadron spin via a COSY matrix.

'TRK' Perform two-chamber track reconstruction.

'TOF' Assign reference points for calculating the relative time-of-flight between the two spectrometers.

$'RES'$ Addition of a gaussian distributed function to a particular Transport ray coordinate.

$'OFF'$ Addition of a constant offset to a particular Transport ray coordinate.

$'H1D'$ A histogram of a Transport coordinate.

$'H2D'$ A three-dimensional histogram of Transport coordinates.

$'SCT'$ A scatter plot of Transport coordinates.

$'NTU'$ An Ntuple of the current Transport vector.

$'CUT'$ A cut on a particular Transport coordinate or pair of coordinates (see below).

<u>$\mathrm{OP} =' \mathrm{HRS}'$</u>

HRS_ID A CHARACTER*1 variable specifying either the JLAB HRS electron arm ("E") or the hadron arm ("H"). Note that this variable may be set independently of the 'C_PID' variable, thus allowing each of the HRS spectrometers to function as either the "electron arm" or the "hadron arm".

APERTURE_TEST A set of five LOGICAL*1 variables, specifying whether each of the five aperture tests should ("T") or should not ("F") be performed. The five aperture tests correspond respectively to: exit of Q1, entrance of DIPOLE, exit of DIPOLE, entrance of Q3 and exit of Q3. The quadrupoles are taken to be circles at both entrance and exit and the dipole is a trapezoid at both entrance and exit. Normally, all five of these variables should be set to "T".

<u>$\mathrm{OP} =' \mathrm{HRI}'$</u>

HRS_ID A CHARACTER*1 variable specifying either the JLAB HRS electron arm ("E") or the hadron arm ("H"). Note that this variable may be set independently of the 'C_PID' variable, thus allowing each of the HRS spectrometers to function as either the "electron arm" or the "hadron arm".

<u>$\mathrm{OP} =' \mathrm{MAT}'$</u>

ORDER An INTEGER variable defining the order of the transformation. Only 1st ("1") and 2nd ("2") order are currently supported.

INVERT A LOGICAL*1 variable ("T" or "F") which indicates whether or not the Transport matrix supplied by the user should first be inverted before its application to the current Transport vector.

TRNSPT_FIL The name of the file which contains the user's Transport matrix. This file must be located in the mceep/dat directory.

$\underline{\mathrm{OP} =' \mathrm{DFT}'}$

DRIFT_LENGTH The drift distance in cm.

$\underline{\mathrm{OP} =' \mathrm{ROT}'}$

ROT_AXIS Rotation axis ($'\mathrm{X}'$, $'\mathrm{Y}'$ or $'\mathrm{Z}'$) which refers to rotations about one of the three Transport axes.

ROT_ANGLE Angle of rotation in degrees. Positive angles imply passive (*i.e.* coordinate system) rotations with a sense given by the right hand rule (*e.g.* for an $'\mathrm{X}'$ rotation, the Y axis is rotated into the direction of the original Z axis).

$\underline{\mathrm{OP} =' \mathrm{POL}'}$

BEND_ANGLE The bend angle for the magnetic element. (This angle should normally be zero for a quadrupole magnet.) For a spectrometer which bends vertically upward, the bend angle should be negative. The sign convention follows the right-hand rule where positive rotations about the spectrometer Y-axis, for instance, imply rotations of the Z-axis into the old X-axis. Note that the $'\mathrm{POL}'$ option calculates spin precession from the bend angle for a given ray. Thus, the $'\mathrm{POL}'$ element must be followed by a $'\mathrm{MAT}'$ element from which the bend angle is determined. (Currently, an $'\mathrm{HRS}'$ element will NOT work with the $'\mathrm{POL}'$ option. For simulations involving the JLAB-Hall A spectrometers it is therefore necessary to use the COSY ($'\mathrm{COS}'$) option instead of $'\mathrm{POL}'$.)

$\underline{\mathrm{OP} =' \mathrm{COS}'}$

COSY_SPIN_FIL The filename for the COSY spin transfer matrix. This file should be contained in the same directory from which MCEEP is run (*i.e.* the directory containing your input). Setting COSY_SPIN_FIL='internal' will cause MCEEP to use a set of standard COSY files (contained in the mceep/dat directory). These files have been obtained for the JLAB-Hall A spectrometers for central proton momenta between 500 and 3500 in steps of 250 MeV/c. (Note that they are specifically for proton ejectiles.) The standard files assume the current particle Transport vector refers to the target. Also, since the COSY spin matrix operates on the polarization vector only, it is necessary to perform any momentum transport *AFTER* the $'$COS$'$ element.

OP $='$ TRK$'$

WC_FIL A CHARACTER variable which specifies the filename containing the wire-chamber parameters. This file must be located in the directory pointed to by the environment variable mceep_dat. For details on the format of this file see the comment header in *get_vect.f*, located in the "sources" directory. Also, an example file is *hrs_vdc.par*, located in the "dat" directory. The $'$TRK$'$ option allows simulation of a pair of wire chambers, where each "chamber" consists of a pair of orthogonal wire planes. Measurement uncertainties, multiple scattering and offsets are taken into account by setting the appropriate parameters in the WC_FIL file.

OP $='$ TOF$'$

CRAY_PATH This is the central ray path length in meters to the current point in the spectrometer line. This identifies the location at which to compute the time-of-flight. For the relative time-of-flight, one must include a $'$TOF$'$ element in each arm. The actual histogram is requested outside of the spectrometer analysis section, as one of the "kinematics" histograms.

OP $='$ RES$'$

OB A CHARACTER*1 variable which specifies the "object" affected by the above operation. Here, object is either $'$K$'$, $'$S$'$, $'$B$'$ or $'$D$'$ (see the text for a complete description of these options):

$OB = 'K'$ The operation affects the Kinematics only. Thus, the cross section is computed from the "unperturbed" values and only the binning intervals for histograms are affected.

$OB = 'S'$ The operation affects the cross section (Sigma) only leaving the binning variables undisturbed.

$OB = 'B'$ The operation affects Both the kinematics and cross sections.

$OB = 'D'$ Default selection. For OP = $'$RES$'$ this is the same as OB = $'$K$'$. (This makes sense when OP = $'$RES$'$ is used to estimate the effects of multiple scattering. In this case the measured kinematics are affected but the cross sections are unchanged since the primary scattering event occurred before the multiple scattering.) However, for OP = 'OFF' this is the same as OB = 'S'. (This is sensible when OP = 'OFF' is used to estimate the effects of a misplaced spectrometer or spectrometer element or detector. In this case since this misplacement is presumably unknown, the kinematics are not affected but a given bin effectively corresponds to a different cross section. This would then correspond to what is actually measured.) See the section on error analysis for further comments.

NCOORD The index of the Transport coordinate to be affected. This value should be 1,2,. . .,6 for x, θ, y, ϕ, l or δ respectively.

RESMOD An integer which specifies whether the following input is to be interpreted as FWHM for the gaussian (RESMOD=1) or as X/X_0 (RESMOD=2), where X/X_0 is the material thickness in radiation lengths. For the latter case, the gaussian sigma is calculated from X/X_0 and from the particle momentum and mass. RESMOD=2 only makes sense for angular coordinates.

RESVAL FWHM for gaussian distributed addition to a particular Transport coordinate (RESMOD=1) or X/X_0 (RESMOD=2). Units for FWHM are in the standard Transport convention (cm, mr or %).

<u>OP =' OFF'</u>

OB A CHARACTER*1 variable which specifies the "object" affected by the above operation. See above (under OP = 'RES') for details.

NCOORD The index of the Transport coordinate to be affected. See above for details.

OFF Offset to a particular Transport coordinate. Units are in the standard Transport convention (cm, mr or %).

<u>OP =' H1D'</u>

IPID_TR Specifies weighting option for the histogram. See the text for a complete description of the various options.

TR_MINX, TR_MAXX Lower and upper limits for the histogram in standard Transport units.

NXCHAN_TR Number of bins for the histogram.

NCUTS_TR Number of Transport cuts to apply to this histogram.

ICUT_IND_TR Indices of the cuts (NCUTS_TR values).

NCOORD Index specifying which Transport coordinate is to be histogrammed. This value should be 1,2,...,6 for x, θ, y, ϕ, l or δ respectively.

TR_FILE Name of the file which will contain the histogram.

$\underline{\mathrm{OP} =' \mathrm{H2D}'}$

IPID_TR Specifies weighting option for the histogram.

TR_MINX, TR_MAXX Lower and upper limits for the X histogram variable in standard Transport units.

TR_MINY, TR_MAXY Lower and upper limits for the Y histogram variable in standard Transport units.

NXCHAN_TR Number of X bins for the histogram.

NYCHAN_TR Number of Y bins for the histogram.

NCUTS_TR Number of Transport cuts to apply to this histogram.

ICUT_IND_TR Indices of the cuts (NCUTS_TR values).

NCOORDX, NCOORDY Indices specifying the X and Y Transport coordinates to be histogrammed. These values should be 1,2,...,6 for x, θ, y, ϕ, l or δ respectively.

TR_FILE Name of the file which will contain the histogram.

$\underline{\mathrm{OP} =' \mathrm{SCT}'}$

TR_MINX, TR_MAXX Lower and upper limits for the X scatter plot variable in standard Transport units.

TR_MINY, TR_MAXY Lower and upper limits for the Y scatter plot variable in standard Transport units.

NCUTS_TR Number of Transport cuts to apply to this scatter plot.

ICUT_IND_TR Indices of the cuts (NCUTS_TR values).

NCOORDX, NCOORDY Indices specifying the X and Y Transport coordinates for the scatter plot. These values should be 1,2,...,6 for x, θ, y, ϕ, l or δ respectively.

TR_FILE Name of the file which will contain the scatter plot.

$\underline{\mathrm{OP} =' \mathrm{NTU}'}$

IPID_TR Specifies weighting option for the Ntuple.

BEND_ANGLE_TOT Total bend angle from the target to the current location in degrees.

TR_FILE Name of the file which will contain the Ntuple.

$\underline{\mathrm{OP} =' \mathrm{CUT}'}$

CUT_DOMAIN CHARACTER*1 variable specifying whether the cut is Globally applied to all histograms (including non-Transport histograms) or applies to a Specific Transport histogram.

CUT_TYPE CHARACTER*1 variable specifying whether the cut is Rectangular or Elliptical.

[For CUT_DOMAIN = 'G' and CUT_TYPE = 'R']

TR_CUT_MIN The minimum value for the cut in standard Transport units.

TR_CUT_MAX The maximum value for the cut in standard Transport units.

NCOORD Index specifying the Transport coordinate for the cut.

[For CUT_DOMAIN = 'G' and CUT_TYPE = 'E']

TR_CUT_X, TR_CUT_Y The length of the semi-major/minor axes for the X and Y coordinates for the cut.

NCOORDX, NCOORDY Indices specifying the X and Y Transport coordinates for the cut.

[For CUT_DOMAIN = 'S' and CUT_TYPE = 'R']

TR_CUT_MIN The minimum value for the cut in standard Transport units.

TR_CUT_MAX The maximum value for the cut in standard Transport units.

NCOORD Index specifying the Transport coordinate for the cut.

TR_CUT_IND Unique index labeling this cut.

[For CUT_DOMAIN = 'S' and CUT_TYPE = 'E']

TR_CUT_X, TR_CUT_Y The length of the semi-major/minor axes for the X and Y coordinates for the cut.

NCOORDX, NCOORDY Indices specifying the X and Y Transport coordinates for the cut.

TR_CUT_IND Unique index labeling this cut.

GLOBAL CUTS

NCUTS The number of global cuts to be applied to all histograms.

ICUT_VAR The index for the variable to be cut.

CUT_MIN The minimum value for the cut.

CUT_MAX The maximum value for the cut.

SPECIFIC CUTS

NCUTS_I The number of histogram specific cuts. These cuts affect only the histograms to which they are attached.

ICUT_IND An identifier referring to a particular cut. This tag is specified in attaching the cut to a particular histogram.

ICUT_VAR_I The index for the variable to be cut.

CUT_MIN_I The minimum value for the cut.

CUT_MAX_I The maximum value for the cut.

KINEMATICS HISTOGRAMS/NTUPLES

NPLOTS The number of histograms and Ntuples to be generated.

PLOT_TYPE A CHARACTER*3 variable denoting the type of plot file to be generated. Options are:

$PLOT\_TYPE = 'P1D'$ Topdrawer histogram.

$PLOT\_TYPE = 'P2D'$ Three-dimensional Topdrawer histogram.

$PLOT\_TYPE = 'SCA'$ Topdrawer scatter plot.

$PLOT\_TYPE = 'NTU'$ MCEEP Ntuple file (single weight).

$PLOT\_TYPE = 'NTM'$ MCEEP Ntuple file (multiple weights).

For PLOT_TYPE = $'$P1D$'$

IPID Index specifying weighting factor for histogram (*i.e.* the dependent variable). Various options include (as described in detail in the text) phase space volumes, (e,e$'$p) yield or average cross section, partial yields for recoil polarization components or average polarizations as well as various singles yields and average cross sections.

X_MIN, X_MAX Minimum and maximum values for independent variable (if left blank, the program will compute them).

NX_CHAN Number of bins for histogram.

X_SC Scale factor for histogram. The default units for histograms are MeV for energies and radians for angles.

X_OFF Offset for histogram. Note that if a histogram is scaled, the offsets reflect the rescaling. Thus, for example, if a histogram is scaled by $180/\pi$ to convert from radians to degrees, the offset is interpreted as degrees.

NCUTS_PL Number of histogram specific cuts to be applied to the histogram.

ICUT_IND_PL(I,J) The tag specifying the cut to be applied to the histogram for J=1 to NCUTS_PL.

L_VAR Index specifying kinematic variable for histogram (independent variable).

PLOT_FIL Filename for the histogram.

For PLOT_TYPE = $'$P2D$'$

IPID Index specifying weighting factor for histogram (*i.e.* the dependent variable).

X_MIN, X_MAX Minimum and maximum values for X variable (if left blank, the program will compute them).

Y_MIN, Y_MAX Minimum and maximum values for Y variable (if left blank, the program will compute them).

NX_CHAN, NY_CHAN Number of X and Y bins for histogram.

X_SC, Y_SC Scale factors for X and Y variables. The default units for histograms are MeV for energies and radians for angles.

X_OFF, Y_OFF Offsets for X and Y variables.

NCUTS_PL Number of histogram specific cuts to be applied to the histogram.

ICUT_IND_PL(I,J) The tag specifying the cut to be applied to the histogram for J=1 to NCUTS_PL.

IX_VAR, IY_VAR Indices specifying X and Y kinematic variables for histogram (independent variables).

PLOT_FIL Filename for the histogram.

<u>For PLOT_TYPE =′ SCA′</u>

X_MIN, X_MAX Minimum and maximum values for X variable (if left blank, the program will compute them).

Y_MIN, Y_MAX Minimum and maximum values for Y variable (if left blank, the program will compute them).

X_SC, Y_SC Scale factors for X and Y variables. The default units are MeV for energies and radians for angles.

X_OFF, Y_OFF Offsets for X and Y variables.

NCUTS_PL Number of histogram specific cuts to be applied to the histogram.

ICUT_IND_PL(I,J) The tag specifying the cut to be applied to the histogram for J=1 to NCUTS_PL.

IX_VAR, IY_VAR Indices specifying X and Y kinematic variables (independent variables).

PLOT_FIL Filename for the scatter plot.

<u>For PLOT_TYPE =′ NTU′</u>

IPID Index specifying weighting factor for Ntuple.

NVAR_NTU Number of coordinates in the Ntuple.

I_VAR(I,J) The indices of the variables in the Ntuple for J=1 to NVAR_NTU.

PLOT_FIL Filename for the Ntuple.

<u>For PLOT_TYPE =′ NTM′</u>

NWT_NTM Number of weight indices to follow.

I_WT(I,J) The indices of the weights in the Ntuple for J=1 to NWT_NTM.

NVAR_NTU Number of coordinates in the Ntuple.

I_VAR(I,J) The indices of the variables in the Ntuple for J=1 to NVAR_NTU.

PLOT_FIL Filename for the Ntuple.

COMMENT LINES

COMMENT(I) Any number of comment lines can appear at the end of the file. The first 10 lines will be written to the summary file.

7.2.3 Sample Input File

A sample input file for MCEEP is shown in Figures 5–6. The case is for d(e,e′p) at $\vec{q} = 1$ GeV/c and $\vec{p}_r = 0$. The sample file does not include any beam smearing or offsets (these values are all set to 0). The collimators are enforced by applying positional cuts at the drift distances corresponding the the front and rear collimator faces. The net drift is subsequently "undone" since the 'HRS' option which follows assumes a starting position at the nominal target center. The electron spectrometer analysis includes the JLAB Hall A HRS option with the electron arm specified (*i.e.* the "electron arm" is being used to detect electrons). All the aperture tests are enabled so that the correct solid angle results. (The hadron arm also specifies the HRS option.) A time-of-flight element is included and also included in the hadron arm so that the relative time-of-flight can be calculated. Note that in the hadron arm the "$-90.$" must be included to establish a vertically bending spectrometer since various spectrometer-frame polarization observables are being histogrammed. For the electron arm, Ntuples at the target before and after resolution effects from the VDCs and spectrometer exit window, *etc.* are output. In addition to Ntuples at the target and focal plane, several "kinematics" histograms and a "kinematics" Ntuple are also requested, some of which have cuts attached. Note that the global cut specified before the kinematics histograms apply to all histograms, even the Transport histograms. Also the spectrometer apertures have the effect of cutting globally. Finally, three comment lines are typed, all of which will appear in the output summary file.

$$\text{Iterations} \begin{cases} 50000 \\ 4,4,4,4,4,4 \end{cases}$$

$$\text{Masses} \left\{ 938.2796, 1, 2.2 \right.$$

$$\text{Kinematics} \left\{ 1653., 0., 0., 1218., 37.0, 0., 1000., -47.1, 0. \right.$$

$$\text{Acceptances} \begin{cases} 4.5, -4.5, 4.5, -4.5 \\ \text{'R','R'}, 60., 130., 60., 130. \end{cases}$$

$$\text{Rate Parameters} \left\{ 126.75, 1., 1. \right.$$

$$\text{Singles Parameters} \left\{ 45., 2.2, 2.2 \right.$$

$$\text{Target Parameters} \begin{cases} 2., 1., 0.169, 2, 1 \\ -0.075, 0.075 \\ 1.109, 1.100 \end{cases}$$

$$\text{Beam Parameters} \begin{cases} 0.8, 0., 0., 0., 0. \\ 0., 0., 0., 0., 0. \\ 0., 0., 0., 0., 0. \\ \text{'R'}, 0.0058, 0.0045 \end{cases}$$

$$\text{Electron Arm Analysis} \begin{cases} \text{'E'}, \mathrm{T}, 21, -90., 0., 0., 0. \\ \text{'NTU'}, 1, 0., \text{'v3\_5\_etgt1.ntu'} \\ \text{'DFT'}, 1.109 \\ \text{'CUT','G','R'}, -6.09, 6.09, 1 \\ \text{'CUT','G','R'}, -3.15, 3.15, 3 \\ \text{'DFT'}, 0.080 \\ \text{'CUT','G','R'}, -6.49, 6.49, 1 \\ \text{'CUT','G','R'}, -3.34, 3.34, 3 \\ \text{'DFT'}, -1.189 \\ \text{'HRS','E'}, \mathrm{T}, \mathrm{T}, \mathrm{T}, \mathrm{T}, \mathrm{T} \\ \text{'NTU'}, 1, 45., \text{'v3\_5\_efp.ntu'} \\ \text{'TOF'}, 23.4 \\ \text{'DFT'}, -0.30 \\ \text{'RES','K'}, 2, 2, 4.0\mathrm{D}-3 \\ \text{'RES','K'}, 4, 2, 4.0\mathrm{D}-3 \\ \text{'DFT'}, 0.15 \\ \text{'RES','K'}, 2, 2, 8.2\mathrm{D}-4 \\ \text{'RES','K'}, 4, 2, 8.2\mathrm{D}-4 \\ \text{'DFT'}, 0.15 \\ \text{'TRK','hrs\_vdc.par'} \\ \text{'HRI,'E'} \\ \text{'NTU'}, 1, 0., \text{'v3\_5\_etgt2.ntu'} \end{cases}$$

Continued in next figure

Figure 5 *Sample Input File for MCEEP – Part I (v3_5.inp).*

Continuation of previous figure

Hadron Arm Analysis

'P', T, 18, −90., 0., 0., 0.
'DFT', 1.100
'CUT', 'G', 'R', −6.09, 6.09, 1
'CUT', 'G', 'R', −3.15, 3.15, 3
'DFT', 0.080
'CUT', 'G', 'R', −6.49, 6.49, 1
'CUT', 'G', 'R', −3.34, 3.34, 3
'DFT', −1.180
'HRS', 'H', T, T, T, T, T
'TOF', 23.4
'DFT', −0.30
'RES', 'K', 2, 2, 4.0D − 3
'RES', 'K', 4, 2, 4.0D − 3
'DFT', 0.15
'RES', 'K', 2, 2, 8.2D − 4
'RES', 'K', 4, 2, 8.2D − 4
'DFT', 0.15
'TRK', 'hrs_vdc.par'
'HRI', 'H'

Global Cuts

1
26, 0., 60.

Specific Cuts

2
1, 15, 410., 460.
2, 16, 975., 1025.

Kinematics Histograms

10
'NTU', 1, 4, 15, 16, 26, 34, 'v3_5.ntu'
'P1D', 0, , , 50, 1., 0., 0, 26, 'v3_5_phsp_pr.top'
'P1D', 1, , , 50, 1., 0., 0, 26, 'v3_5_yield_pr.top'
'P1D', −1, , , 50, 1., 0., 0, 26, 'v3_5_sig_pr.top'
'P1D', −1, , , 50, 1., 0., 2, 1, 2, 26, 'v3_5_sig_pr_c.top'
'P2D', 1, , , , , 25, 25, 1., 1., 0., 0., 0, 15, 16, 'v3_5_yield_wq.top'
'SCA', , , , , 1., 1., 0., 0., 0, 15, 16, 'v3_5_phsp_wq.top'
'P1D', 6, , , 50, 1., 0., 0, 12, 'v3_5_tpoly_pf.top'
'P1D', −6, , , 50, 1., 0., 0, 12, 'v3_5_tpol_pf.top'
'P1D', 1, −50., −10., 50, 1., 0., 0, 30, 'v3_5_tof.top'

Comment Lines

Comments : d(e, e'p)n with JLAB Hall A HRS
50 uA on 15 cm LD2
External radiation included

Figure 6 *Sample Input File for MCEEP – Part II (v3_5.inp).*

Chapter 8

PHYSICS MODELS OF MCEEP

Currently, MCEEP provides a menu of several basic physics models. Effort has been made to allow inclusion of new and hopefully more sophisticated models with ease. The next chapter describes the steps needed to add a new physics model. This chapter is devoted to a description of the models MCEEP currently employs.

8.1 General Formalism of (e,e′N)

A detailed analysis of the (e,e′N) reaction has been developed by Picklesimer and Van Orden.[17] Their analysis shows that there are 18 independent response functions in the general case, for a polarized electron beam, involving nucleon polarization and non-coplanar kinematics:

$$\begin{aligned}\frac{d^3\sigma}{d\omega d\Omega_e d\Omega_p} = \frac{m|\vec{p}'|}{2(2\pi)^3}\left[\frac{d\sigma}{d\Omega_e}\right]_{Mott} &\times \big\{v_L(R_L + R_L^n S_n) + v_T(R_T + R_T^n S_n) \\ &+ v_{TT}\big[(R_{TT} + R_{TT}^n S_n)\cos 2\phi_x + (R_{TT}^l S_l + R_{TT}^t S_t)\sin 2\phi_x\big] \\ &+ v_{LT}\big[(R_{LT} + R_{LT}^n S_n)\cos\phi_x + (R_{LT}^l S_l + R_{LT}^t S_t)\sin\phi_x\big] \\ &+ h v_{LT'}\big[(R_{LT'} + R_{LT'}^n S_n)\sin\phi_x + (R_{LT'}^l S_l + R_{LT'}^t S_t)\cos\phi_x\big] \\ &+ h v_{TT'}(R_{TT'}^l S_l + R_{TT'}^t S_t)\big\}.\end{aligned} \tag{62}$$

The response functions, R, depend on $\vec{q}$, ω, T_p (the proton kinetic energy) and θ_{pq} (the nucleon emission angle with respect to $\vec{q}$). Here, ϕ_x is the angle between the electron scattering plane and the plane containing $\vec{q}$ and the detected nucleon. The v's are known kinematic factors weighting the various virtual photon polarization states and $[d\sigma/d\Omega_e]_{Mott}$ is the cross section for scattering from a structureless Dirac particle. The S's are the three components of a unit vector pointing along the nucleon spin direction.

The response functions arise from products of various components of the nuclear current and are therefore expected to exhibit varying sensitivities to different aspects of the reaction. Measurements involving separation of these response functions will consequently provide selectivity to various dynamical models of the (e,e′N) reaction. Conversely, under certain assumptions, these response functions can be calculated and the results used to simulate gross features of (e,e′N) experiments. In the following section, the form of the (e,e′N) cross section in the simplest nontrivial calculation is given.

8.2 (e,e′N) in PWIA (Options 100 and 200)

MCEEP's menu option numbers 100 and 200 provide for evaluation of the coincidence (e,e′N) cross section in PWIA (Plane Wave Impulse Approximation). Under option 100, MCEEP computes only the four unpolarized response functions. Option 200 provides calculations for all response functions which are not zero in PWIA. Both options produce the same results for the unpolarized cross section. The information provided by option 200 is therefore a superset of that for option 100. However, if only unpolarized observables are desired, option 100 is preferred since it requires less computation time.

8.2.1 Factorization of the Cross Section

In PWIA a simple form for the (e,e′N) cross section arises. In this approximation the cross section factors into an elementary electron-nucleon off-shell cross section times a quantity which contains the nuclear structure information, the spectral function:[18]

$$\sigma_{eeN} = K\sigma_{eN}\frac{1}{\eta}\alpha_{sf}S(\vec{p}_r, \epsilon_m) \tag{63}$$

where K is a kinematical factor. The recoil factor, η, is taken as unity for scattering to the continuum of the residual system and for scattering to a bound state it is given by

$$\eta = 1 - \frac{E_p}{E_r}\frac{\vec{p}_p \cdot \vec{p}_r}{|\vec{p}_p|^2} \tag{64}$$

where $E_{p(r)}$ is the total energy of the detected nucleon (recoiling system) and $\vec{p}_p$ is the nucleon final momentum. The spectral function, S, represents the joint probability for finding within the nucleus a nucleon of momentum $-\vec{p}_r$ and separation energy ϵ_m. The spectral function calculated by MCEEP also contains the overall multiplicative (spectroscopic) factor, α_{sf}. This factor is specified in the user input file. All of the dependence on virtual photon polarization is contained within the eN cross section. Many ambiguities exist in the extrapolation of the on-shell cross section to that for a bound nucleon. However, all prescriptions are fairly consistent for nucleons which are nearly on-shell. MCEEP uses the "CC1" prescription of de Forest for σ_{eN}.[19] MCEEP provides a variety of nuclear spectral functions; the various choices are described in the next subsection. In most cases the spectral function itself is assumed to factorize:

$$S(\vec{p}_r, \epsilon_m) = |\phi(p_r)|^2 f(\epsilon_m). \tag{65}$$

For a bound state the missing mass distribution function, $f(\epsilon_m)$, is a delta function. In this case the missing mass integral leads to the Jacobian given above (η) and the cross section becomes 5-fold differential instead of 6-fold. Because of the difference in dimensionality (*i.e.* the effective constraint implied by the fixed missing mass for the bound state case), MCEEP treats the bound state problem and the continuum problem separately. For the continuum case the missing mass distribution function must be supplied to MCEEP via a file (see the next subsection).

8.2.2 Spectral Functions

MCEEP provides a variety of spectral functions to use with the factorized models for the (e,e′N) reaction. The various options are described below. The deuteron, ^{3}He, ^{4}He, ^{16}O and ^{208}Pb momentum distributions are displayed in Figures 7, 8, 9, 10 and 11 respectively. The integrals of each momentum distribution used by MCEEP are shown in Table 2. (Note that these are the integrals before application of the spectroscopic factors contained in the input file, SPEC_FAC. In some cases, the momentum distributions already reflect measured spectroscopic factors.)

Note that if the user selects the continuum case and a spectral function option which only specifies the p_r dependence, a missing mass distribution file will also be requested by the program.

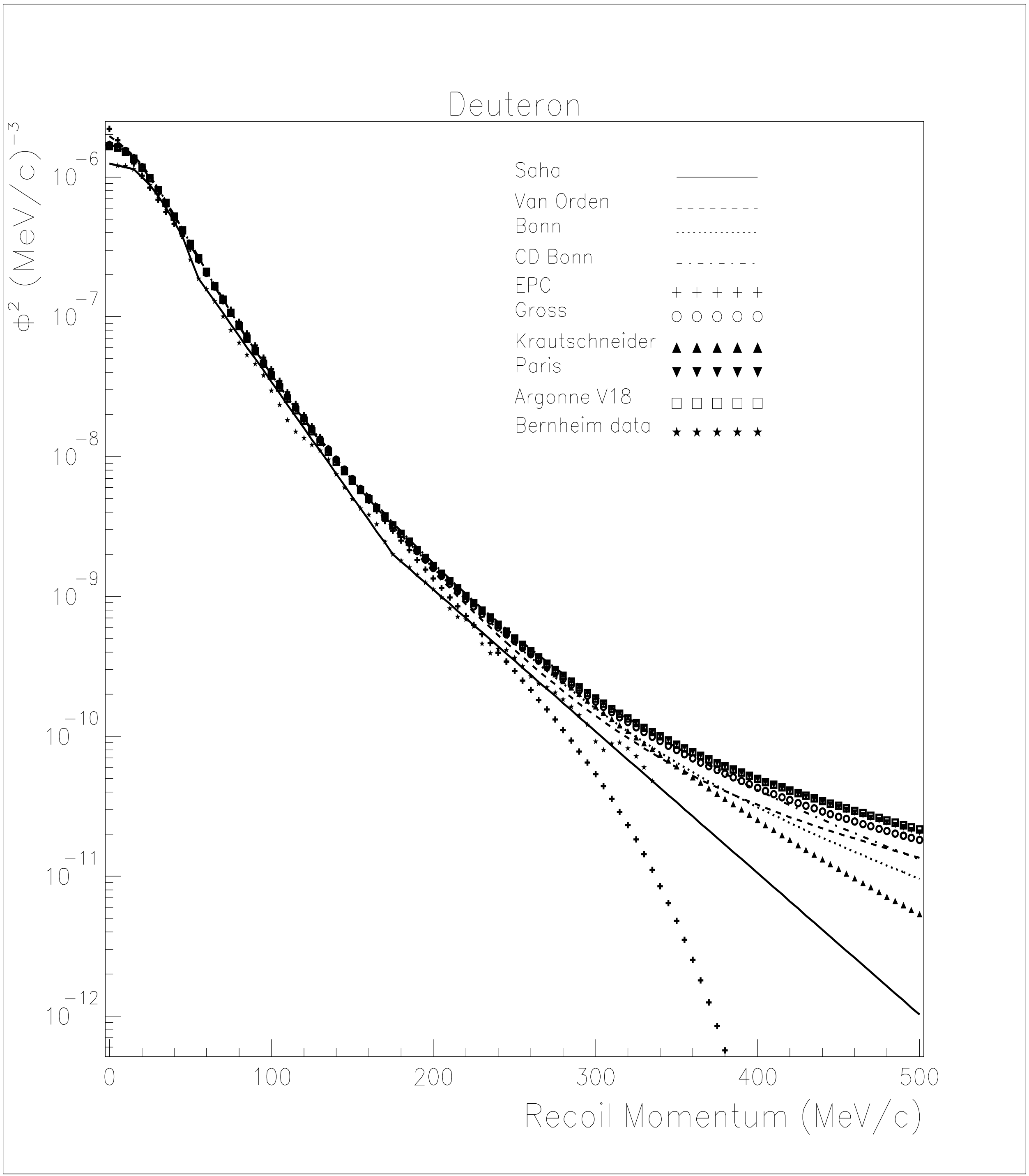

Figure 7. *Momentum distributions employed by MCEEP for d(e,e′p)n. The Argonne V18 and Paris distributions are almost indistinguishable. See the text for details regarding the various distributions.*

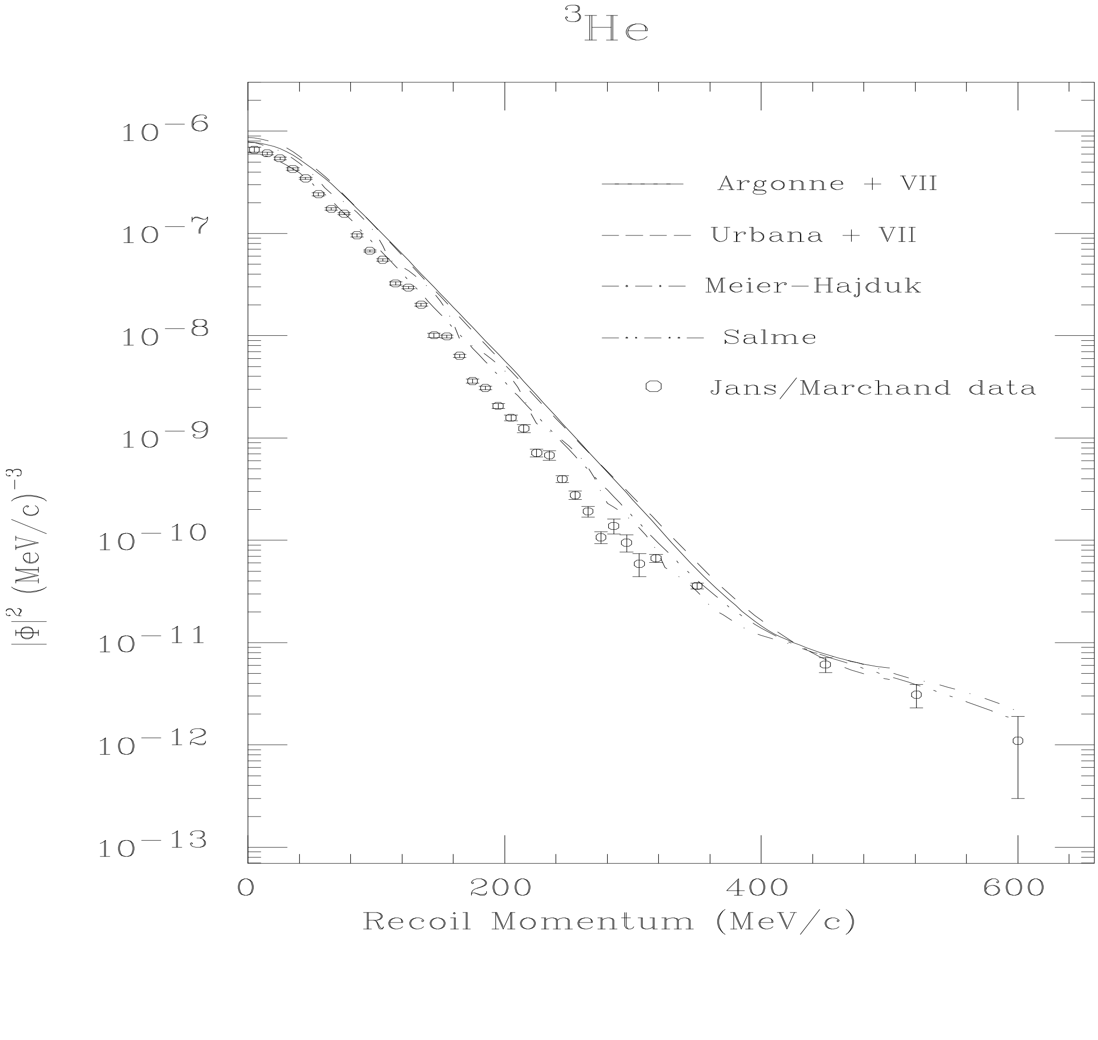

Figure 8. *Momentum distributions employed by MCEEP for the two-body breakup channel of ^{3}He (i.e. ^{3}He(e,e$'$p)d). See the text for details regarding the various distributions.*

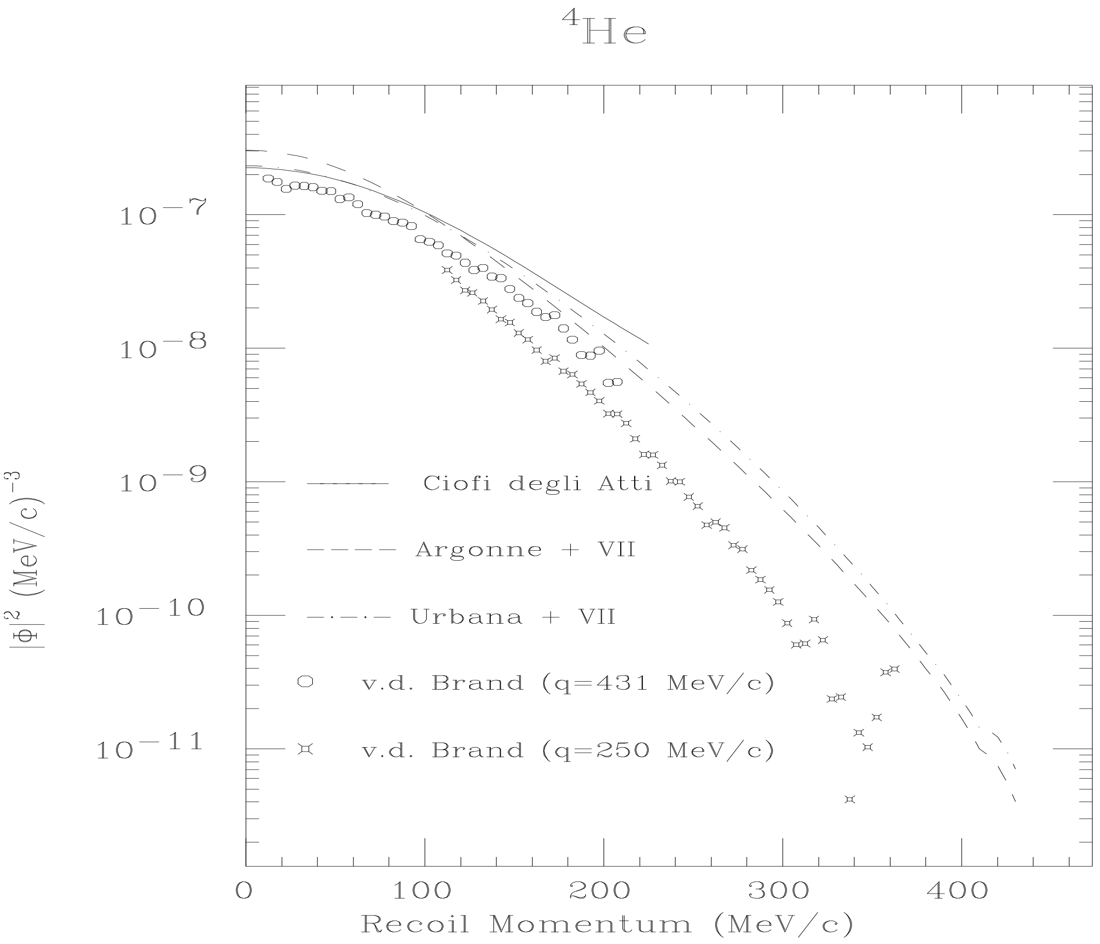

Figure 9. *Momentum distributions employed by MCEEP for the two-body breakup channel of ^{4}He (i.e. ^{4}He(e,e$'$p)t). See the text for details regarding the various distributions.*

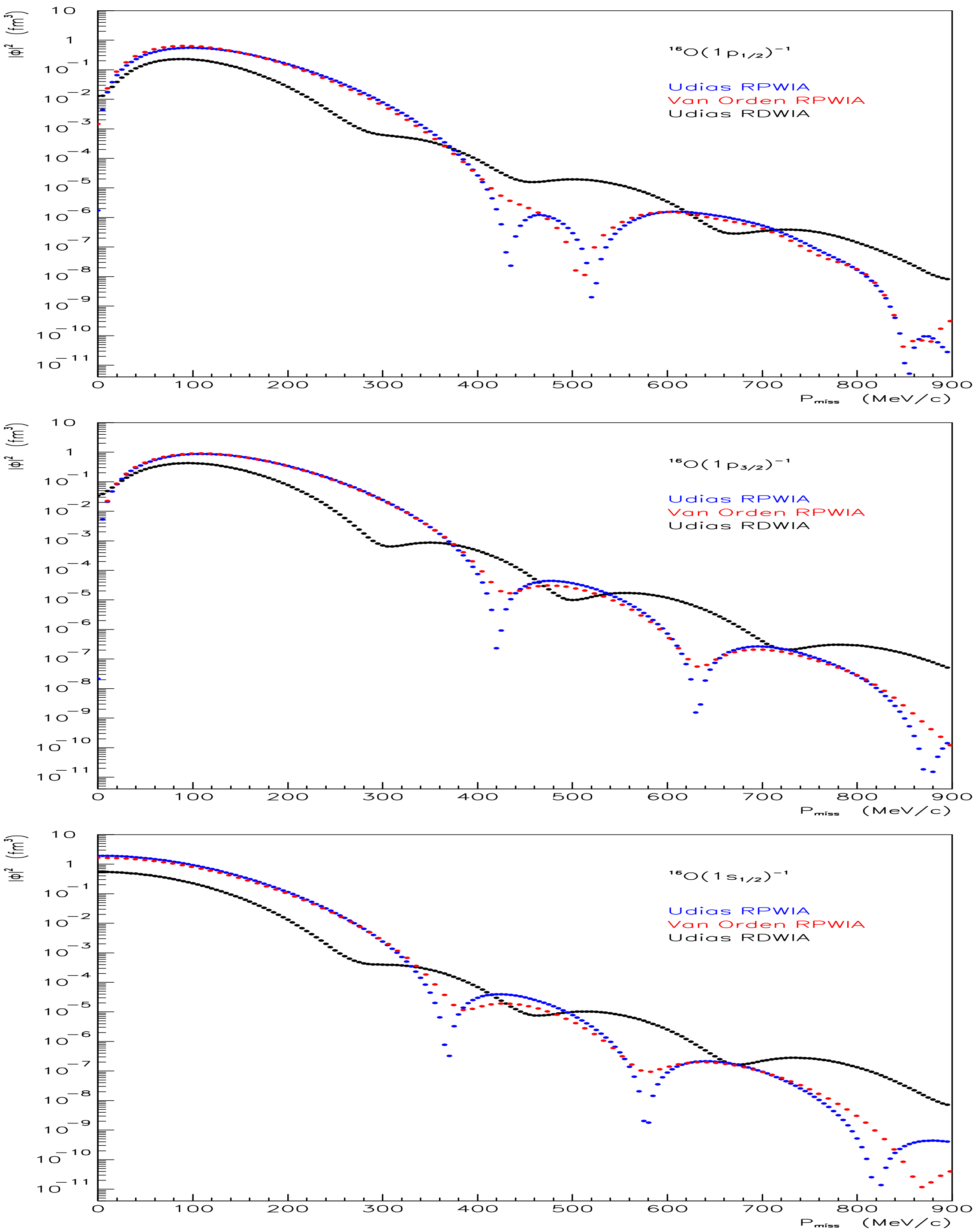

Figure 10. *Momentum distributions employed by MCEEP for ^{16}O. The distributions are normalized to $2J+1$ and for DWIA include both FSI and the measured spectroscopic factors. See the text for details.*

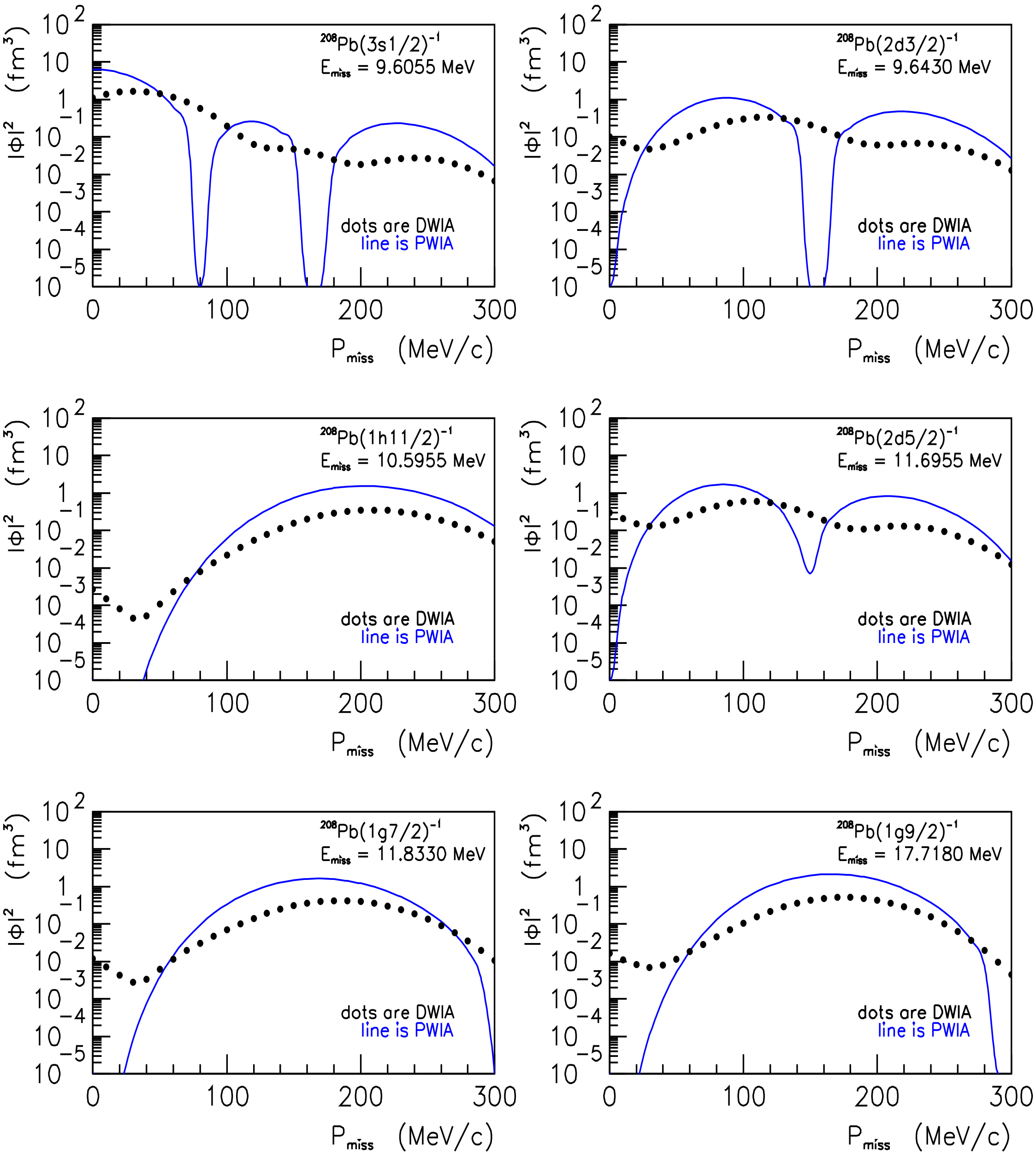

Figure 11. *DWIA momentum distributions from Lapikas employed by MCEEP for ^{208}Pb. The distributions are normalized to $2J+1$ and include both FSI and the measured spectroscopic factors. See the text for details.*

Option #	Integration Range (MeV/c)	Integral
10	0 to ∞	1
11	0 to ∞	0.810
12	0 to ∞	0.936
13	0 to ∞	0.776
14	0 to ∞	1
15–19	0 to ∞	1
20 (2-body)	0 to ∞	1.62
20 (3-body)	0 to ∞	1.06
21	0 to 500	2
22	0 to 500	2
23	0 to ∞	1.07
24 (2-body)	0 to ∞	1.33
24 (3-body)	0 to ∞	0.64
30	0 to 225	2
31	0 to 430	1.92
32	0 to 430	2
33	0 to 207.5	1.139
34	112.5 (0) to 362.5	0.405 (0.998)
40	0 to ∞	2
41	0 to ∞	4
42	0 to ∞	2
43	0 to 895	0.530
44	0 to 895	1.199
45	0 to 895	0.462
80	0 to 300	0.766
81	0 to 300	1.215
82	0 to 300	2.882
83	0 to 300	1.932
84	0 to 300	2.648
85	0 to 300	2.825
150–158	0 to ∞	1

Table 2 *Integrals for MCEEP momentum distributions:* $4\pi \int dp_r p_r^2 |\phi(p_r)|^2$ *(2-body) or* $4\pi \int d\epsilon_m dp_r p_r^2 S(p_r, \epsilon_m)$ *(3-body). The integration range for* p_r *is indicated. For the 3-body distributions the* ϵ_m *integration range is over the entire range for which the spectral function is defined. Note that in some cases the experimental distribution is extrapolated to yield the integral shown.*

Unit Weighting

1: This option sets the momentum distribution to unity. In the continuum case, the user will be prompted to provide a missing mass distribution as well. This option is useful as a divisor for data since, to the extent the remainder of the model is correct, the result will be the experimental momentum distribution.

Deuteron

10: This parameterization is due to Krautschneider.[20] The integral is forced to one in the MCEEP implementation.

11: This is a parametric fit to the data of Bernheim (see option 13) by A. Saha.[21]

12: This parameterization is used by the program EPC of Lightbody and O'Connell[16], the routines of which are currently used in MCEEP to generate singles cross sections.

13: Experimental distribution measured by Bernheim *et al.*[22]

14: A parameterization by J.W. Van Orden.[23]

15: The Paris parameterization by M. Lacombe *et al.*[24]

16: The Bonn parameterization.[25]

17: The Argonne V18 parameterization.[25]

18: The CD Bonn parameterization.[25]

19: The Gross parameterization.[25]

^{3}He

20: This is the distribution of Meier-Hajduk *et al.*[26] If the continuum case is selected the 3-body breakup part of the distribution is used, otherwise the 2-body breakup distribution is used.

21: This parameterization is from a variational calculation using the Argonne potential for the 2-body breakup of ^{3}He.[27]

22: This parameterization is from a variational calculation using the Urbana potential for the 2-body breakup of ^{3}He.[27]

23: This is an experimental distribution from Jans *et al.*[28] and Marchand *et al.*[29] These data are for the 2-body break-up channel only. Also, the data for p_r=155–195 MeV/c are averages of the Kinematics I and Kinematics II data sets.

24: This is the distribution of Kievsky *et al.*[30] If the continuum case is selected the 3-body breakup part of the distribution is used, otherwise the 2-body breakup distribution is used.

^{4}He

30: This is a parameterization from Ciofi degli Atti.[31] Note that for $p_r > 225$ MeV/c this parameterization yields unphysical results. Therefore, the momentum distribution is set to zero above this value.

31: This parameterization is for the t+p breakup channel of ^{4}He using the Argonne potential.[27] The parameterization becomes unreasonable for $p_r > 430$ MeV/c and so the momentum distribution is set to zero in this range.

32: This parameterization is for the t+p breakup channel of ^{4}He using the Urbana potential.[27] The parameterization becomes unreasonable for $p_r > 430$ MeV/c and so the momentum distribution is set to zero in this range.

33: This is the experimental momentum distribution from van den Brand *et al.*[32] for the t+p breakup channel of ^{4}He measured at $\vec{q} = 431$ MeV/c. Note that this data set extends only to $p_r = 207.5$ MeV/c. Therefore, the momentum distribution is set to zero above this value.

34: This is the experimental momentum distribution from van den Brand *et al.*[32] for the t+p breakup channel of ^{4}He measured at $\vec{q} = 250$ MeV/c. Note that this data set extends from $p_r = 112.5$ to 362.5 MeV/c. Extrapolations give fairly reasonable values for $p_r < 112.5$ MeV/c (within a factor of two of the $\vec{q} = 431$ MeV/c data set). However, extrapolation beyond 362.5 MeV/c yields unreasonable values. Therefore, the momentum distribution is set to zero above this range.

^{16}O

All distributions are normalized to $2J + 1$. (The J. Udias parameterizations were coded by K. Fissum.)

40: The PWIA parameterization of J.W. Van Orden for the $1p_{1/2}$ shell.[23]

41: The PWIA parameterization of J.W. Van Orden for the $1p_{3/2}$ shell.[23]

42: The PWIA parameterization of J.W. Van Orden for the $1s_{1/2}$ shell.[23]

43: The RDWIA parameterization of J. Udias for the $1p_{1/2}$ shell.[33] The distribution (contained in the file mceep/dat/16o_ju_dw.dat) is specifically for parallel kinematics with $e = 4050$ MeV and $T_p = 432$ MeV. This distribution reflects both FSI and the measured spectroscopic factors.

44: The RDWIA parameterization of J. Udias for the $1p_{3/2}$ shell.[33] The distribution (contained in the file mceep/dat/16o_ju_dw.dat) is specifically for parallel kinematics with $e = 4050$ MeV and $T_p = 432$ MeV. This distribution reflects both FSI and the measured spectroscopic factors.

45: The RDWIA parameterization of J. Udias for the $1s_{1/2}$ shell.[33] The distribution (contained in the file mceep/dat/16o_ju_dw.dat) is specifically for parallel kinematics with $e = 4050$ MeV and $T_p = 432$ MeV. This distribution reflects both FSI and the measured spectroscopic factors.

$\boxed{^{208}\mathrm{Pb}}$

These CDWIA parameterizations are From L. Lapikas[34] and were coded by K. Fissum. The distributions (contained in the file mceep/dat/208pb_11_dw.dat) are specifically for parallel kinematics with $e = 674$ MeV and $T_p = 160$ MeV. All distributions are normalized to $2J + 1$ and reflect both FSI and the measured spectroscopic factors.

80: The DWIA parameterization by L. Lapikas of ^{208}Pb for the $3s_{1/2}$ shell (spectroscopic factor = 0.5694).

81: The DWIA parameterization by L. Lapikas of ^{208}Pb for the $2d_{3/2}$ shell (spectroscopic factor = 0.6372).

82: The DWIA parameterization by L. Lapikas of ^{208}Pb for the $1h_{11/2}$ shell (spectroscopic factor = 0.7732).

83: The DWIA parameterization by L. Lapikas of ^{208}Pb for the $2d_{5/2}$ shell (spectroscopic factor = 0.7052).

84: The DWIA parameterization by L. Lapikas of ^{208}Pb for the $1g_{7/2}$ shell (spectroscopic factor = 0.7898).

85: The DWIA parameterization by L. Lapikas of ^{208}Pb for the $1g_{9/2}$ shell (spectroscopic factor = 0.7846).

Harmonic Oscillator

MCEEP provides a set of generic harmonic oscillator momentum distributions. If one of these is selected the user must also enter the oscillator parameter (in fm). Options 150–158 correspond respectively to the 1s, 1p, 2s, 1d, 2p, 1f, 3s, 2d and 1g shells.

External Files

These options allow the use of externally supplied spectral distributions. The distribution must be represented by an array (contained in a file) on which MCEEP can interpolate. The exact format required is specified in the file "spectral.for".

160: Two-dimensional spectral function $S(p_r, \epsilon_m)$.

161: Momentum distribution $|\phi(p_r)|^2$ for a bound state.

162: Momentum distribution and missing mass distribution, $f(\epsilon_m)$, for the continuum case.

8.3 Relativistic d(e,e′p)n (Option 300)

MCEEP provides a fully relativistic calculation for the d(e,e′p)n reaction in the Impulse Approximation. This calculation employs the formalism of Dmitrasinovic and Gross.[35] Note that this option is explicitly for (e,e′p). Note also that these routines, although they appear to give reasonable results for both the cross section and proton polarizations, have not been fully tested.

8.4 Relativisitic PWBA for d(e,e′p)n (Option 350)

[This section was contributed by Sabine Jeschonnek.]

This code for D(e,e′p)n is based on the calculations published by S. Jeschonnek and T. W. Donnelly.[36] The only difference to the implementation in MCEEP is that the paper deals only with PWIA (*i.e.* the photon couples only to the proton), while the code also offers the option to use PWBA (*i.e.* the photon can also couple to the neutron).

The electromagnetic one-body current operator employed here does not contain any non-relativistic reduction. It was derived from the current operator for a free nucleon, and then the popular "on-shell" ansatz was used for the description of the off-shell behavior (for details see paper). This operator works very well; for a comparison with a fully covariant calculation, see S. Jeschonnek and J. W. Van Orden.[37]

The code has options to use the full relativistic version of the current operator, a first-order relativistic version, which in most cases will be very close to the full version, and a strict non-relativistic version. This non-relativistic current operator will in general lead to very different results and its use is not recommended for the description/simulation of actual data/count rates.

Both the particular form of non-relativistic wave-function and the form-factor parametrization can be chosen. The code has Paris, Bonn, CD Bonn and Argonne V18 wave functions, and a variety of form factors: dipole, dipole+Galster, several MMD versions and a parametrization by Wally Van Orden which includes even the latest data from JLab.

The code does not contain any FSI (available for some kinematic regions, but takes too long for a Monte Carlo), no meson exchange currents (available in a non-relativistic form, they also take too long), and no isobar states.

For questions and comments, contact Sabine Jeschonnek (sabine@jlab.org), or Bill Donnelly (donnelly@mitlns.mit.edu).

8.5 Pion Electroproduction in p(e,e′p)π^0 (Option 500)

This option computes cross section and recoil polarization observables for pion electroproduction on the proton in p(e,e′p)π^0 using the routines of R.W. Lourie[38] with modifications by G.A. Warren.[39] It uses the helicity amplitude formalism and includes the nucleon pole (Born) terms. The amplitudes for the Δ and higher resonances are taken from the dispersion relation based parameterization of Devenish and Lyth.[40] The user may also employ a set of external Roper amplitudes supplied via files. If Option 500 is selected, the user must supply the name of a file which contains various resonance options and parameters. A "standard" resonance option file is located in mceep/dat and is called *res_standard.dat*.

8.6 Pion Electroproduction in p(e,e′π^+)n (Option 600)

This option calculates cross sections in the p(e,e′π^+)n reaction using the routines of G.M. Huber[41] with modifications by L. Qin.[42] The cross sections are calculated from a fit to the data of Brauel *et al.*[43] with the large Q^2 parameterization of Bebek.[44] The pion survival probability is taken into account via a weighting factor for the cross section. In order for this to work, a ′TOF′ element must be included in the hadron spectrometer input section so that the flight path is known.

8.7 Kaon Electroproduction in p(e,e′K$^+$)Λ (Option 700)

This option calculates cross sections in the p(e,e′K$^+$)Λ reaction using routines of P. Markowitz[45] based on a prescription of Saghai.[46] The kaon survival probability is taken into account via a weighting factor for the cross section. In order for this to work, a ′TOF′ element must be included in the hadron spectrometer input section so that the flight path is known.

8.8 Arenhövel Responses in d(e,e′p)n (Option 804)

This option reads in a set of files for the Arenhövel unpolarized response functions in d(e,e′p)n. The ratio of the resulting cross section to PWIA is linearly interpolated and then scaled to the PWIA result at the exact kinematics. This allows a coarser grid, since the ratio is likely to be flatter than the cross section itself. The Arenhövel files should first be processed with the script "aren_response.prl" located in the mceep/utilities directory.

Chapter 9

MODIFYING THE PROGRAM

MCEEP was written in a way that allows the user to modify the program with minimal effort. Of course, not all potential implementations of MCEEP could be anticipated. However, special attention has been payed to the two areas which will likely need to be specifically tailored for any user. These are: adding new histogram variables and incorporating new physics models. The required changes to the program for each of these cases is now described.

9.1 Adding a New Histogram Variable

All histogram variables are calculated in the subroutine KINEM. This routine is called in the calculation of the default variable ranges (subroutine RANGES) as well as in the main Monte Carlo loop (the main routine, MCEEP). Thus, new variables need only to be added to subroutine KINEM. So that the other routines have access to the new variable the Fortran INCLUDE file, *var.cmn*, must be modified. In addition, the file *var_dat.cmn* in which the label for the variable is specified must also be updated. (These files are located in the mceep/cmn directory.) There are four steps that need to be completed (if there is any confusion just look at one of the existing variables in the *var.cmn* file and *var_dat.cmn* file and do likewise):

"var.cmn"

i) Declare the new variable as DOUBLE PRECISION.

ii) Equivalence the new variable to the next element of the array VAR. When requesting histograms or placing cuts within the input file, the variable is referenced via the index in this array.

"var_dat.cmn"

iii) Update NUM_VAR in the DATA statement to reflect the new number of variables.

iv) Choose a name for the variable and add this to the list of DATA statements for the array VAR_NAME. This name is only needed for the output summary file so that the formatted output displays a mnemonic for each variable.

Note: Make sure the new variables are added to the *end* of the list. It is suggested that the existing variables are not modified or removed since some of them are used in the calculation of cross sections. In addition, the formatted output produced by subroutine SUMMARY requires that the indices of the existing variables not change, since these indices are referenced directly in one of the WRITE statements. Finally, it is suggested that all subroutines which include *var.cmn* or *var_dat.cmn) be recompiled.*

In summary, the following steps must be taken as described above:

1) Add the calculation of the new variable in subroutine KINEM.

2) Modify the include file *var.cmn* as described in steps i) and ii) above.

3) Modify the include file *var_dat.cmn* as described in steps iii) and iv) above.

4) Recompile all routines which include *var.cmn* or *var_dat.cmn* and relink using make.

9.2 Incorporating New Physics Models

There are two subroutines which need to be changed to add a new physics model, PHYS_CHOICE and PHYSICS.

PHYS_CHOICE is called by MCEEP before the Monte Carlo event loop. This routine allows the user to select the physics model at run-time and it also performs any preliminaries such as opening and reading files. For instance, if one of the factorized (e,e′N) models is chosen, PHYS_CHOICE calls the subroutine SPECT_SETUP which then asks the user to select a spectral function so that other preliminaries may be performed. Thus, the first step in incorporating a new physics option is to add the option to the list of selections in PHYS_CHOICE. As is done for the existing options, any required files should be opened and read within this routine. The new option must have a unique selection number associated with it, which will be referenced by the variable, IPHYS_OPT.

All cross sections (other than single-arm) and polarizations are returned by the subroutine PHYSICS, which is called within the main event loop in MCEEP. This routine must be modified so that the user's physics subroutine is called under the appropriate IPHYS_OPT number. The new physics routine should return SIGMA_EEP (the coincidence cross section) as well as POL_N, POL_L and POL_T (the parts of the cross section which correspond to the three ejectile spin orientations) and ASYMMETRY (the electron analyzing power) if these observables are calculated. Note that the "polarizations" computed by the new user routine are not polarizations in the usual sense; rather they are the pieces of the *cross section* which correspond to the three spin orientations. MCEEP will calculate the polarizations from these automatically by dividing these arrays by the unpolarized cross section array, channel by channel, if the user requests histograms of average polarization within the input file. For polarization Ntuples, this division is done outside of MCEEP using the KUMAC file, *mc.kumac*, located in the directory mceep/utilities. (The MCEEP Ntuple must first be converted to a PAW Ntuple using *mceep_util* in the same directory.) The quantities computed by the physics routine should have units of

$\mathbf{fm^2/(sr^2\text{-}MeV)}$ for a **bound** state (5-fold differential) or $\mathbf{fm^2/(sr^2\text{-}MeV\text{-}MeV/c)}$ for **continuum** states (6-fold differential). (Note that the 6-fold cross section must be differential in the hadron momentum, not the hadron energy, since it is the momentum which is randomly sampled.) In computing the yields, MCEEP multiplies these cross sections by the appropriate "SIGFACTOR" (depending on whether it is bound state or continuum). The units of SIGFACTOR are sr^2-MeV/fm^2 (bound state) or sr^2-MeV-(MeV/c)/fm^2 (continuum). Thus, the yields are dimensionless as required.

If the user wishes to add a new spectral function for one of the existing factorized (e,e′N) models, only the routines SPECT_SETUP and SPECTRAL need to be modified. In SPECT_SETUP the new option should be added to the list of choices and a unique index should be associated with the choice as referenced by the variable ISPEC_OPT. Also, any necessary preliminaries such as opening and reading files should be performed by calling a user routine from SPECT_SETUP. Then a piece should be added to the routine SPECTRAL so that the new spectral function is calculated every event. The units of the function SPECTRAL should be $\mathbf{(MeV/c)^{-3}}$ for a **bound** state or $\mathbf{(MeV/c)^{-3}}$-$\mathbf{(MeV)^{-1}}$ for the **continuum** case.

In summary, the following steps should be taken to add new physics models or spectral functions to MCEEP:

New Physics Model

1) Add option to the list in PHYS_CHOICE. Associate unique integer identifier (IPHYS_OPT) with the new option.

2) Perform preliminaries (open files, *etc.*) in PHYS_CHOICE.

3) Perform calculation of SIGMA_EEP and optionally POL_N, POL_L, POL_T and ASYMMETRY by calling a user subroutine from subroutine PHYSICS. These quantities should have units of $fm^2/(sr^2$-MeV) (for bound state) or of $fm^2/(sr^2$-MeV-MeV/c) (for continuum).

New Spectral Function

1) Add option to the list in SPECT_SETUP. Associate unique integer identifier with the new option (ISPEC_OPT).

2) Perform preliminaries (open files, *etc.*) in SPECT_SETUP.

3) Perform calculation of SPECTRAL by calling a user subroutine within function SPECTRAL. SPECTRAL should have units of $(MeV/c)^{-3}$ (bound state) or $(MeV/c)^{-3}$-$(MeV)^{-1}$ (continuum).

Appendix A DESCRIPTION OF KINEMATIC VARIABLES

MCEEP allows for histograms of any of the weighting functions (singles cross sections, coincidence cross sections, phase space, *etc.*) vs. various kinematic variables. In addition MCEEP can produce Ntuples of these kinematic variables where one of the elements in the Ntuple is the appropriate weight. Although the user can easily add kinematic variables to the list, none of the standard ones should be removed. This is because some of these variables are used by the various cross section calculation routines. A table of the standard kinematic variables follows. Each variable is then described in detail.

Index	**Variable**	**Default Units**
1	x_e	meters
2	θ_e	radians
3	y_e	meters
4	ϕ_e	radians
5	z_e	meters
6	p_e	MeV/c
7	x_p	meters
8	θ_p	radians
9	y_p	meters
10	ϕ_p	radians
11	z_p	meters
12	p_p	MeV/c
13	E_0	MeV
14	θ_s	radians
15	ω	MeV
16	$\lvert\vec{q}\rvert$	MeV/c
17	θ_q	radians
18	ϕ_q	radians
19	q_μ^2	(GeV/c)^2
20	x	dimensionless
21	ϵ	dimensionless
22	θ_{pq}	radians
23	ϕ_x	radians
24	ϵ_m	MeV
25	p_r	MeV/c
26	$\lvert\vec{p}_r\rvert$	MeV/c
27	f_{rec}	dimensionless
28	W	MeV

29	θ_{CM}	radians
30	$\Delta\tau$	nsec
31	$\theta_{p_r q}$	radians
32	Γ	dimensionless
33	β_h	dimensionless
34	E_m	MeV
35	${\epsilon_m}^2$	MeV^2
36	p_{rx}	MeV/c
37	p_{ry}	MeV/c
38	p_{rz}	MeV/c
39	ΔE_b	MeV
40	ΔE_e	MeV
41	ΔE_p	MeV
42	$< \Delta E_b >$	MeV
43	$< \Delta E_e >$	MeV
44	$< \Delta E_p >$	MeV
45	x_v	meters
46	y_v	meters
47	z_v	meters
48	θ_{pr}	radians
49	ϕ_{pr}	radians
51	S_{nn}	dimensionless
52	S_{nt}	dimensionless
53	S_{nl}	dimensionless
54	S_{tn}	dimensionless
55	S_{tt}	dimensionless
56	S_{tl}	dimensionless
57	S_{ln}	dimensionless
58	S_{lt}	dimensionless
59	S_{ll}	dimensionless
61	x^e_{tg}	cm
62	θ^e_{tg}	mr
63	y^e_{tg}	cm
64	ϕ^e_{tg}	mr
65	l^e_{tg}	cm
66	δ^e_{tg}	%
71	x^p_{tg}	cm
72	θ^p_{tg}	mr
73	y^p_{tg}	cm

74	ϕ^p_{tg}	mr
75	l^p_{tg}	cm
76	δ^p_{tg}	%
81	x^e_{fp}	cm
82	θ^e_{fp}	mr
83	y^e_{fp}	cm
84	ϕ^e_{fp}	mr
85	l^e_{fp}	cm
86	δ^e_{fp}	%
91	x^p_{fp}	cm
92	θ^p_{fp}	mr
93	y^p_{fp}	cm
94	ϕ^p_{fp}	mr
95	l^p_{fp}	cm
96	δ^p_{fp}	%
101	x^e_{fin}	cm
102	θ^e_{fin}	mr
103	y^e_{fin}	cm
104	ϕ^e_{fin}	mr
105	l^e_{fin}	cm
106	δ^e_{fin}	%
111	x^p_{fin}	cm
112	θ^p_{fin}	mr
113	y^p_{fin}	cm
114	ϕ^p_{fin}	mr
115	l^p_{fin}	cm
116	δ^p_{fin}	%
117	E^v_0	MeV
118	p^v_e	MeV/c
119	ω^v	MeV
120	$\lvert\vec{q}^{\,v}\rvert$	MeV/c
121	$(q^2_\mu)^v$	$(\mathrm{GeV/c})^2$
122	ϕ^v_x	radians
123	E^v_m	MeV
124	ϵ^v_m	MeV
125	$\lvert\vec{p_r}^{\ v}\rvert$	MeV/c
126	W^v	MeV
127	θ^v_{CM}	radians
128	θ^v_{pq}	radians

129	θ_s^v	radians

Index: 1
Variable: x_e
Fortran Name: BEAM_E1
Default Units: meters
Description: The x coordinate of the origination point within the target of the scattered electron referred to the "modified laboratory" coordinate system (for definition of the laboratory system see the text). Here "modified" refers to a shift of the origin to correspond with the spectrometer object point. This allows for spectrometer mispointing.

Index: 2
Variable: θ_e
Fortran Name: TH_E_I
Default Units: radians
Description: The vertical angle of the scattered electron at the target referred to the "laboratory" coordinate system (see the text).

Index: 3
Variable: y_e
Fortran Name: BEAM_E2
Default Units: meters
Description: Same as BEAM_E1, except for the y coordinate.

Index: 4
Variable: ϕ_e
Fortran Name: PH_E_I
Default Units: radians
Description: The horizontal angle of the scattered electron at the target referred to the "laboratory" coordinate system (see the text).

Index: 5
Variable: z_e
Fortran Name: BEAM_E3
Default Units: meters
Description: Same as BEAM_E1, except for the z coordinate.

Index: 6
Variable: p_e
Fortran Name: PF_E_I
Default Units: MeV/c
Description: The momentum of the scattered electron.

Index: 7
Variable: x_p
Fortran Name: BEAM_P1
Default Units: meters
Description: The x coordinate of the origination point within the target of the emerging hadron referred to the "modified laboratory" coordinate system (for definition of the laboratory system see the text). Here "modified" refers to a shift of the origin to correspond with the spectrometer object point. This allows for spectrometer mispointing. Note that this is not necessarily the same as BEAM_E1.

Index: 8
Variable: θ_p
Fortran Name: TH_P_I
Default Units: radians
Description: The vertical angle of the emerging hadron at the target referred to the "laboratory" coordinate system (see the text).

Index: 9
Variable: y_p
Fortran Name: BEAM_P2
Default Units: meters
Description: Same as BEAM_P1, except for the y coordinate.

Index: 10
Variable: ϕ_p
Fortran Name: PH_P_I
Default Units: radians
Description: The horizontal angle of the emerging hadron at the target referred to the "laboratory" coordinate system (see the text).

Index: 11
Variable: z_p
Fortran Name: BEAM_P3
Default Units: meters
Description: Same as BEAM_P1, except for the z coordinate.

Index: 12
Variable: p_p
Fortran Name: PF_P_I
Default Units: MeV/c
Description: The momentum of the emerging hadron.

Index: 13
Variable: E_0
Fortran Name: E0_I
Default Units: MeV
Description: The energy of the incident electron after offsets and resolution smearing of the beam.

Index: 14
Variable: θ_s
Fortran Name: TSCAT
Default Units: radians
Description: The electron scattering angle.

Index: 15
Variable: ω
Fortran Name: OMEGA
Default Units: MeV
Description: The electron energy transfer.

Index: 16
Variable: $|\vec{q}\,|$
Fortran Name: QMAG
Default Units: MeV/c
Description: The magnitude of the 3-momentum transfer.

Index: 17
Variable: θ_q
Fortran Name: THEQ
Default Units: radians
Description: The vertical angle of $\vec{q}$ with respect to the "laboratory" coordinate system.

Index: 18
Variable: ϕ_q
Fortran Name: PHIQ
Default Units: radians
Description: The horizontal angle of $\vec{q}$ with respect to the "laboratory" coordinate system.

Index: 19
Variable: q_μ^2
Fortran Name: QMU2_G
Default Units: $(\mathrm{GeV/c})^2$
Description: The square of the electron 4-momentum transfer.

Index: 20
Variable: x
Fortran Name: X
Default Units: dimensionless
Description: The Bjorken x variable, $x = q_\mu^2/2m\omega$.

Index: 21
Variable: ϵ
Fortran Name: EPSILON
Default Units: dimensionless
Description: The longitudinal polarization of the virtual photon ($0 \leq \epsilon \leq 1$), given by: $\epsilon = 1/(1 + 2|\vec{q}|^2 \tan^2(\theta_s/2)/q_\mu^2)$.

Index: 22
Variable: θ_{pq}
Fortran Name: THETA_PQ
Default Units: radians
Description: The angle between the emerging hadron and $\vec{q}$.

Index: 23
Variable: ϕ_x
Fortran Name: PHIX
Default Units: radians
Description: The out-of-plane angle (*i.e.* the angle between the electron scattering plane and the plane containing $\vec{q}$ and the emerging hadron).

Index: 24
Variable: ϵ_m
Fortran Name: MISS_M
Default Units: MeV
Description: The missing mass (*i.e.* excitation energy of the residual nuclear system).

Index: 25
Variable: p_r
Fortran Name: PREC
Default Units: MeV/c
Description: The momentum of the recoiling nuclear system. This quantity is negative if the recoil momentum has a component along $\vec{q}$: $p_r = -\vec{p}_r \cdot \vec{q}/|\hat{p}_r \cdot \vec{q}\,|$.

Index: 26
Variable: $|\vec{p}_r\,|$
Fortran Name: PRMAG
Default Units: MeV/c
Description: The magnitude of the recoil momentum.

Index: 27
Variable: f_{rec}
Fortran Name: RECFAC
Default Units: dimensionless
Description: The recoil factor. This is a Jacobian which is applied for bound states so that the resulting cross section is five-fold differential instead of six-fold (an integration over the bound state missing mass results in this Jacobian).

Index: 28
Variable: W
Fortran Name: W
Default Units: MeV
Description: The invariant mass of the hadronic final state.

Index: 29
Variable: θ_{CM}
Fortran Name: THETA_CM
Default Units: radians
Description: The angle of the ejectile with respect to $\vec{q}$ in the center-of-mass system.

Index: 30
Variable: $\Delta\tau$
Fortran Name: TOF_REL
Default Units: nsec
Description: The relative time of flight between electron and hadron.

Index: 31
Variable: $\theta_{p_r q}$
Fortran Name: THETA_PRQ
Default Units: radians
Description: The angle of the recoiling system with respect to $\vec{q}$.

Index: 32
Variable: Γ
Fortran Name: SURVIVE_PROB
Default Units: dimensionless
Description: The survival probability for physics options 600 (π^+) and 700 (K^+).

Index: 33
Variable: β_h
Fortran Name: BETA_H
Default Units: dimensionless
Description: The ejectile velocity in units of c.

Index: 34
Variable: E_m
Fortran Name: EMISS
Default Units: MeV
Description: The missing energy defined by $E_m = \omega - T_p$. Note this is NOT the missing mass (see variable 24).

Index: 35
Variable: ${\epsilon_m}^2$
Fortran Name: MMSQ
Default Units: MeV^2
Description: The square of the missing mass (NOT missing energy). Note: this is a signed quantity.

Index: 36
Variable: p_{rx}
Fortran Name: PREC_X
Default Units: MeV/c
Description: The x component of $\vec{p}_r$ in the "laboratory" coordinate system.

Index: 37
Variable: p_{ry}
Fortran Name: PREC_Y
Default Units: MeV/c
Description: The y component of $\vec{p}_r$ in the "laboratory" coordinate system.

Index: 38
Variable: p_{rz}
Fortran Name: PREC_Z
Default Units: MeV/c
Description: The z component of $\vec{p}_r$ in the "laboratory" coordinate system.

Index: 39
Variable: ΔE_b
Fortran Name: ELOSS_B_I
Default Units: MeV
Description: The energy loss of the beam.

Index: 40
Variable: ΔE_e
Fortran Name: ELOSS_E_I
Default Units: MeV
Description: The energy loss of the scattered electron.

Index: 41
Variable: ΔE_p
Fortran Name: ELOSS_P_I
Default Units: MeV
Description: The energy loss of the ejectile.

Index: 42
Variable: $< \Delta E_b >$
Fortran Name: DE_DX_CORR_B
Default Units: MeV
Description: The mean energy loss of the beam. The energy is "corrected" by this amount if ELOSS_MOD=2.

Index: 43
Variable: $< \Delta E_e >$
Fortran Name: DE_DX_CORR_E
Default Units: MeV
Description: The mean energy loss of the scattered electron. The energy is "corrected" by this amount if ELOSS_MOD=2.

Index: 44
Variable: $< \Delta E_p >$
Fortran Name: DE_DX_CORR_P
Default Units: MeV
Description: The mean energy loss of the ejectile. The energy is "corrected" by this amount if ELOSS_MOD=2.

Index: 45
Variable: x_v
Fortran Name: VERTEX_X
Default Units: meters
Description: The X coordinate of the beam-target interaction vertex referred to the "laboratory" coordinate system.

Index: 46
Variable: y_v
Fortran Name: VERTEX_Y
Default Units: meters
Description: The Y coordinate of the beam-target interaction vertex referred to the "laboratory" coordinate system.

Index: 47
Variable: z_v
Fortran Name: VERTEX_Z
Default Units: meters
Description: The Z coordinate of the beam-target interaction vertex referred to the "laboratory" coordinate system.

Index: 48
Variable: θ_{pr}
Fortran Name: PREC_TH
Default Units: radians
Description: The vertical angle of the recoil momentum vector in the "laboratory" coordinate system.

Index: 49
Variable: ϕ_{pr}
Fortran Name: PREC_PH
Default Units: radians
Description: The horizontal angle (relative to the nominal beam direction) of the recoil momentum vector in the "laboratory" coordinate system.

Index: 51
Variable: S_{nn}
Fortran Name: VAR(51) or PMATRIX(1,1)
Default Units: dimensionless
Description: The nn spin transfer matrix element for the ejectile.

Index: 52
Variable: S_{nt}
Fortran Name: VAR(52) or PMATRIX(1,2)
Default Units: dimensionless
Description: The nt spin transfer matrix element for the ejectile.

Index: 53
Variable: S_{nl}
Fortran Name: VAR(53) or PMATRIX(1,3)
Default Units: dimensionless
Description: The nl spin transfer matrix element for the ejectile.

Index: 54
Variable: S_{tn}
Fortran Name: VAR(54) or PMATRIX(2,1)
Default Units: dimensionless
Description: The tn spin transfer matrix element for the ejectile.

Index: 55
Variable: S_{tt}
Fortran Name: VAR(55) or PMATRIX(2,2)
Default Units: dimensionless
Description: The tt spin transfer matrix element for the ejectile.

Index: 56
Variable: S_{tl}
Fortran Name: VAR(56) or PMATRIX(2,3)
Default Units: dimensionless
Description: The tl spin transfer matrix element for the ejectile.

Index: 57
Variable: S_{ln}
Fortran Name: VAR(57) or PMATRIX(3,1)
Default Units: dimensionless
Description: The ln spin transfer matrix element for the ejectile.

Index: 58
Variable: S_{lt}
Fortran Name: VAR(58) or PMATRIX(3,2)
Default Units: dimensionless
Description: The lt spin transfer matrix element for the ejectile.

Index: 59
Variable: S_{ll}
Fortran Name: VAR(59) or PMATRIX(3,3)
Default Units: dimensionless
Description: The ll spin transfer matrix element for the ejectile.

Index: 61
Variable: x^e_{tg}
Fortran Name: VAR(61)
Default Units: cm
Description: Transport x for the electron before the spectrometer analysis (*i.e.* at the target).

Index: 62
Variable: θ^e_{tg}
Fortran Name: VAR(62)
Default Units: mr
Description: Transport θ for the electron before the spectrometer analysis (*i.e.* at the target).

Index: 63
Variable: y^e_{tg}
Fortran Name: VAR(63)
Default Units: cm
Description: Transport y for the electron before the spectrometer analysis (*i.e.* at the target).

Index: 64
Variable: ϕ^e_{tg}
Fortran Name: VAR(64)
Default Units: mr
Description: Transport ϕ for the electron before the spectrometer analysis (*i.e.* at the target).

Index: 65
Variable: l^e_{tg}
Fortran Name: VAR(65)
Default Units: cm
Description: Transport l for the electron before the spectrometer analysis (*i.e.* at the target).

Index: 66
Variable: δ^e_{tg}
Fortran Name: VAR(66)
Default Units: %
Description: Transport δ for the electron before the spectrometer analysis (*i.e.* at the target).

Index: 71
Variable: x^p_{tg}
Fortran Name: VAR(71)
Default Units: cm
Description: Transport x for the hadron before the spectrometer analysis (*i.e.* at the target).

Index: 72
Variable: θ_{tg}^{p}
Fortran Name: VAR(72)
Default Units: mr
Description: Transport θ for the hadron before the spectrometer analysis (*i.e.* at the target).

Index: 73
Variable: y_{tg}^{p}
Fortran Name: VAR(73)
Default Units: cm
Description: Transport y for the hadron before the spectrometer analysis (*i.e.* at the target).

Index: 74
Variable: ϕ_{tg}^{p}
Fortran Name: VAR(74)
Default Units: mr
Description: Transport ϕ for the hadron before the spectrometer analysis (*i.e.* at the target).

Index: 75
Variable: l_{tg}^{p}
Fortran Name: VAR(75)
Default Units: cm
Description: Transport l for the hadron before the spectrometer analysis (*i.e.* at the target).

Index: 76
Variable: δ_{tg}^{p}
Fortran Name: VAR(76)
Default Units: %
Description: Transport δ for the hadron before the spectrometer analysis (*i.e.* at the target).

Index: 81
Variable: x_{fp}^{e}
Fortran Name: VAR(81)
Default Units: cm
Description: Transport x for the electron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 82
Variable: θ^e_{fp}
Fortran Name: VAR(82)
Default Units: mr
Description: Transport θ for the electron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 83
Variable: y^e_{fp}
Fortran Name: VAR(83)
Default Units: cm
Description: Transport y for the electron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 84
Variable: ϕ^e_{fp}
Fortran Name: VAR(84)
Default Units: mr
Description: Transport ϕ for the electron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 85
Variable: l^e_{fp}
Fortran Name: VAR(85)
Default Units: cm
Description: Transport l for the electron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 86
Variable: δ^e_{fp}
Fortran Name: VAR(86)
Default Units: %
Description: Transport δ for the electron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 91
Variable: x^p_{fp}
Fortran Name: VAR(91)
Default Units: cm
Description: Transport x for the hadron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 92
Variable: θ^p_{fp}
Fortran Name: VAR(92)
Default Units: mr
Description: Transport θ for the hadron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 93
Variable: y^p_{fp}
Fortran Name: VAR(93)
Default Units: cm
Description: Transport y for the hadron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 94
Variable: ϕ^p_{fp}
Fortran Name: VAR(94)
Default Units: mr
Description: Transport ϕ for the hadron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 95
Variable: l^p_{fp}
Fortran Name: VAR(95)
Default Units: cm
Description: Transport l for the hadron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 96
Variable: δ^p_{fp}
Fortran Name: VAR(96)
Default Units: %
Description: Transport δ for the hadron at the focal plane (*i.e.* after the 'HRS' element in the spectrometer input section).

Index: 101
Variable: x^e_{fin}
Fortran Name: VAR(101)
Default Units: cm
Description: Transport x for the electron after the spectrometer analysis (*i.e.* normally at the target).

Index: 102
Variable: θ^e_{fin}
Fortran Name: VAR(102)
Default Units: mr
Description: Transport θ for the electron after the spectrometer analysis (*i.e.* normally at the target).

Index: 103
Variable: y^e_{fin}
Fortran Name: VAR(103)
Default Units: cm
Description: Transport y for the electron after the spectrometer analysis (*i.e.* normally at the target).

Index: 104
Variable: ϕ^e_{fin}
Fortran Name: VAR(104)
Default Units: mr
Description: Transport ϕ for the electron after the spectrometer analysis (*i.e.* normally at the target).

Index: 105
Variable: l^e_{fin}
Fortran Name: VAR(105)
Default Units: cm
Description: Transport l for the electron after the spectrometer analysis (*i.e.* normally at the target).

Index: 106
Variable: δ^e_{fin}
Fortran Name: VAR(106)
Default Units: %
Description: Transport δ for the electron after the spectrometer analysis (*i.e.* normally at the target).

Index: 111
Variable: x^p_{fin}
Fortran Name: VAR(111)
Default Units: cm
Description: Transport x for the hadron after the spectrometer analysis (*i.e.* normally at the target).

Index: 112
Variable: θ^p_{fin}
Fortran Name: VAR(112)
Default Units: mr
Description: Transport θ for the hadron after the spectrometer analysis (*i.e.* normally at the target).

Index: 113
Variable: y^p_{fin}
Fortran Name: VAR(113)
Default Units: cm
Description: Transport y for the hadron after the spectrometer analysis (*i.e.* normally at the target).

Index: 114
Variable: ϕ^p_{fin}
Fortran Name: VAR(114)
Default Units: mr
Description: Transport ϕ for the hadron after the spectrometer analysis (*i.e.* normally at the target).

Index: 115
Variable: l^p_{fin}
Fortran Name: VAR(115)
Default Units: cm
Description: Transport l for the hadron after the spectrometer analysis (*i.e.* normally at the target).

Index: 116
Variable: δ^p_{fin}
Fortran Name: VAR(116)
Default Units: %
Description: Transport δ for the hadron after the spectrometer analysis (*i.e.* normally at the target).

Index: 117
Variable: E^v_0
Fortran Name: E0_V
Default Units: MeV
Description: The energy of the incident electron corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 118
Variable: p_e^v
Fortran Name: EF_V
Default Units: MeV/c
Description: The momentum of the scattered electron corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 119
Variable: ω^v
Fortran Name: OMEGA_V
Default Units: MeV
Description: The electron energy transfer corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 120
Variable: $|\vec{q}^{\,v}|$
Fortran Name: QMAG_V
Default Units: MeV/c
Description: The magnitude of the 3-momentum transfer corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 121
Variable: $(q_\mu^2)^v$
Fortran Name: QMU2_V
Default Units: $(\mathrm{GeV/c})^2$
Description: The square of the electron 4-momentum transfer corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 122
Variable: ϕ_x^v
Fortran Name: PHIX_V
Default Units: radians
Description: The out-of-plane angle (*i.e.* the angle between the electron scattering plane and the plane containing $\vec{q}$ and the emerging hadron) corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 123
Variable: E_m^v
Fortran Name: EMISS_V
Default Units: MeV
Description: The missing energy defined by $E_m = \omega - T_p$. Note this is NOT the missing mass

Index: 124
Variable: ϵ_m^v
Fortran Name: MISSM_V
Default Units: MeV
Description: The missing mass (*i.e.* excitation energy of the residual nuclear system) corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 125
Variable: $|\vec{p_r}\,^v|$
Fortran Name: PRMAG_V
Default Units: MeV/c
Description: The magnitude of the recoil momentum corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 126
Variable: W^v
Fortran Name: W_V
Default Units: MeV
Description: The invariant mass of the hadronic final state corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 127
Variable: θ_{CM}^v
Fortran Name: THCM_V
Default Units: radians
Description: The angle of the ejectile with respect to $\vec{q}$ in the center-of-mass system corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 128
Variable: θ_{pq}^{v}
Fortran Name: THPQ_V
Default Units: radians
Description: The angle between the emerging hadron and $\vec{q}$ corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Index: 129
Variable: θ_{s}^{v}
Fortran Name: TSCAT_V
Default Units: radians
Description: The electron scattering angle corresponding to the reaction "vertex" (as opposed to the directly measurable "asymptotic" value which is affected by radiation).

Appendix B MCEEP UTILITIES

A separate utilities program, *mceep_util*, can be found in the directory mceep/utilities. The main function of this program is to convert MCEEP output files to HBOOK files. Having this functionality in an external program avoids the need to link MCEEP with any libraries. The utility program currently has three options:

1) *MCEEP_TO_HBOOK:* Converts MCEEP 1-D and 2-D Topdrawer output and MCEEP Ntuple output to HBOOK format. A script file (*make_hbook* in mceep/utilities) can be used to generate a single HBOOK file from all the MCEEP output (except for scatter plots). This script scans the MCEEP input file for requested plots and Ntuples, processes each of these files in turn using the MCEEP_TO_HBOOK routine and produces a single HBOOK file. The script is executed by a command: "make_hbook [input_file_prefix]", where "[input_file_prefix]" is the name of the MCEEP input file prefix (*i.e.* omitting the ".inp").

2) *RATIO:* Produces a ratio or difference histogram from two 1-D MCEEP generated Topdrawer files.

3) *GAUSS FILE:* Produces a gaussian distributed weighting function (normalized to unity). The resulting file can be used as a missing mass distribution function for a bound state, for example.

Appendix C PORTING MCEEP

The official MCEEP release now runs on five platforms: DEC-Ultrix, DEC-Alpha (OSF1), HP-UX, SunOS and Linux (with the Absoft f77 compiler and with the GNU g77 compiler). All of the MCEEP files are common across these systems including the source code. The distribution and instructions for porting mceep may be obtained from:

http://www.physics.odu.edu/ũlmer/mceep/mceep.html

The required files and procedure to port the program will now be described.

1) Get the self-extracting archive, *mceep_v3_5.uu*, and place this file into your MCEEP home directory. (As of Version 3.1 MCEEP can be built from any directory.) During the procedure (described below) a directory tree will be created off this directory: $mceep_home/mceep.

2) Type: "chmod u+x mceep_v3_5.uu" (without the quotes).

3) Type: "./mceep_v3_5.uu". This will extract the uuencoded, gzipped tar file, unuuencode it, unzip it and untar it. It will also execute the "build" script to produce makefiles and then make the mceep program as well as the mceep_util program.

4) The build script will inquire which operating system you are porting to. Answer with one of the listed possibilities.

After attempting this port, if the utility program, *mceep_util*, does not link successfully, you may need to edit the file mceep/utilities/makefile so that the CERN libraries are properly referenced.

Various directories will be created when you build MCEEP. They are:

Source directories

Unix names:	mceep/sources and mceep/gross
Contains:	All Fortran sources/objects (executable is in mceep/sources)

Input directory

Unix name:	mceep/input
Contains:	Sample input file and resulting output from MY run of MCEEP

Output directory

Unix name:	mceep/output

Contains:	This will contain YOUR output after you execute the compare_v3_5 script

Data directory

Unix name:	mceep/dat
Contains:	Parameter files, random number seeds, *etc.*

Common directory

Unix name:	mceep/cmn
Contains:	The common block modules for Fortran sources

Utilities directory

Unix name:	mceep/utilities
Contains:	The MCEEP companion program, *mceep_util*

Distribution directory

Unix name:	mceep/distribute
Contains:	The MCEEP distribution kit

A sample input file called *v3_5.inp* should be located in the input directory. The results obtained by MCEEP using *v3_5.inp* as input (for the bound state case in (e,e′p) with radiative effects, multiphoton correction, cutoff of 1 MeV and with physics option 200 and spectral function 10) are located in the input directory as well. These files provide a comparison so that you can verify that the program is working properly. To verify the port of mceep, change directory to mceep/utilities and execute the script: *compare_v3_5*. This script will run mceep with the *v3_5.inp* input file and compare the output generated with that from your distribution.

Acknowledgements

I would like to express my appreciation to Mina Nozar for her work on the spectrometer analysis routines, to Douglas Higinbotham, Garth Huber, Robert Lourie, Pete Markowitz, Liming Qin, Scott Van Verst and Glen Warren for providing some of the physics routines, to Franck Sabatie for work on implementation of the beam raster, to Luminita Todor for work on the radiative effects, to Steffen Strauch for work on the scattering plane polarization observables and to Sabine Jeschonnek for providing her PWBA d(e,e′p)n model and documentation. I thank K. Fissum for providing code and figures for the ^{16}O spectral functions from Udias and for the ^{208}Pb spectral functions from Lapikas. I also thank Mark Jones for work on the schwinger correction routine and for incorporating COSY spin transport and Joe Mitchell for providing the perl script to process the Arenhövel response function files. Finally, I thank Mike Finn for many useful discussions.

References

[1] K.L. Brown, F. Rothacher, D.C. Carey and C.H. Iselin, **TRANSPORT**, *A Computer Program for Designing Charged Particle Beam Transport Systems*, SLAC-91, Rev. 2, UC-28(1/A).

[2] R. Brun, O. Couet, C. Vandoni and P. Zanarini, *PAW Users Guide*, Program Library Q121, CERN (1991).

[3] R. Brun, *HBOOK Users Guide (Version 4.15)*, Program Library Y250, CERN (1992).

[4] I thank Jeff Templon for pointing this out. Versions of MCEEP prior to 3.0 neglected this, which can lead to significant error for very long targets.

[5] J. Schwinger, Phys. Rev. **75**, 898 (1949).

[6] S. Penner, Nuclear Structure Physics, Proceedings of the 18th Scottish Univ. Summer School in Physics (1977).

[7] L.W. Mo and Y.S. Tsai, Rev. Mod. Phys. **41**, 205 (1969).

[8] E. Borie and D. Drechsel, Nucl. Phys. **A167**, 369 (1971).

[9] J.A. Templon, C.E. Vellidis, R.E.J. Florizone and A.J. Sarty, Phys. Rev. C **61**, 014607 (2000).

[10] William R. Leo, *Techniques for Nuclear and Particle Physics Experiments*, Springer-Verlag, 1994.

[11] R.M. Sternheimer, Atomic Data and Nuclear Data Tables **30**, 261–271 (1984).

[12] G.R. Lynch and O.I. Dahl, Nucl. Inst. Meth. in Phys. Res. **B58**, 6–10 (1991).

[13] J.A. Templon, *Radiative-Tail Simulation for Mainz Expt. A1/1-93*, October 1998.

[14] A. Picklesimer and J.W. Van Orden, Phys. Rev. C **35**, 266 (1987).

[15] J.A. Tjon and M.J. Zuilhof, Phys. Lett. **84B**, 31 (1979); M.J. Zuilhof and J.A. Tjon, Phys. Rev. C **22**, 2369 (1980); J.A. Tjon, private communication.

[16] J.W. Lightbody and J.S. O'Connell, Computers in Physics, May/June 1988, p. 57.

[17] A. Picklesimer and J.W. Van Orden, Phys. Rev. C **40**, 290 (1989).

[18] S. Frullani and J. Mougey, *Advances in Nuclear Physics*, vol. 14, ed. by J.W. Negele and E. Vogt, Plenum Press (1985).

[19] T. de Forest, Jr., Nucl. Phys. **A392**, 232 (1983).

[20] F. Krautschneider, Ph.D. Thesis, Bonn University, BONN-IR-76-37 (1976); L. Hulthen and M. Sagawara, Handb. d. Phys. Bd. 39, S.1.

[21] A. Saha, private communication.

[22] M. Bernheim *et al.*, Nucl. Phys. **A365**, 349 (1981).

[23] J.W. Van Orden, private communication.

[24] M. Lacombe *et al.*, Phys. Lett. **101B**, 139 (1981).

[25] S. Jeschonnek, private communication.

[26] H. Meier-Hajduk *et al.*, Nucl. Phys. **A395**, 332 (1983) and private communication.

[27] R. Schiavilla, V.R. Pandharipande and R.B. Wiringa, Nucl. Phys. **A449**, 219 (1986) and private communication.

[28] E. Jans *et al.*, Phys. Rev. Lett. **49**, 974 (1982).

[29] C. Marchand *et al.*, Phys. Rev. Lett. **60**, 1703 (1988).

[30] A. Kievsky, E. Pace, G. Salme and M. Viviani, Phys. C **56**, 64 (1997) and J. Templon, private communication (1999).

[31] C. Ciofi degli Atti, private communication.

[32] J.F.J. van den Brand *et al.*, Phys. Rev. Lett. **60**, 2006 (1988) and private communication.

[33] J. Udias, private communication to K. Fissum.

[34] L. Lapikas, private communication to K. Fissum.

[35] V. Dmitrasinovic and Franz Gross, Phys. Rev. C **40**, 2479 (1989) and J.W. Van Orden, private communication.

[36] S. Jeschonnek and T. W. Donnelly, Phys. Rev. C **57**, 2438 (1998).

[37] S. Jeschonnek and J. W. Van Orden, Phys. Rev. C **62**, 044613 (2000).

[38] R.W. Lourie, Nucl. Phys. **A509**, 653 (1990) and private communication.

[39] G.A. Warren, private communication (1997).

[40] R.C.E. Devenish and D.H. Lyth, Nucl. Phys. **B43**, 228 (1972).

[41] G.M. Huber, private communication (1996).

[42] L. Qin, private communication (1996).

[43] P. Brauel *et al.*, Z. Phys. C3, 101 (1979).

[44] C.J. Bebek *et al.*, Phys. Rev. D **17**, 1693 (1978).

[45] P. Markowitz, private communication (1999).

[46] J.C. David *et al.*, Phys. Rev. C **53**, 2613 (1996).