N69-17750

NASA CR-72467

KINEMATIC RELATIONS AND PHASE SPACE

FOR THE REACTION M(x, xt)N

by P. C. Gugelot

Prepared under Contract No. NAS 3-12178

September 1968

for Lewis Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NASA CR-72467

TOPICAL REPORT

KINEMATIC RELATIONS AND PHASE SPACE FOR THE REACTION M(x, xt)N

by

P. C. Gugelot
Virginia Associated Research Center
Newport News, Virginia 23606

prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

September 1968

CONTRACT NAS 3-12178

Technical Management
NASA Lewis Research Center
Cleveland, Ohio
Physics and Chemistry Division
John S. Vincent

E-4718

KINEMATIC RELATIONS AND PHASE SPACE

FOR THE REACTION M(x, x t)N

by P. C. Gugelot*

I. Introduction

The description of the interaction of an incoming particle with a particle bound in a nuclear system in terms of purely relativistically invariant kinematic relations offers several problems. In general a heuristic description is given in the following way. Fig. 1 presents an incoming nucleon x colliding with a nucleon t of the Fermi-sea in the potential U . B is the binding energy of the particle t in the potential.

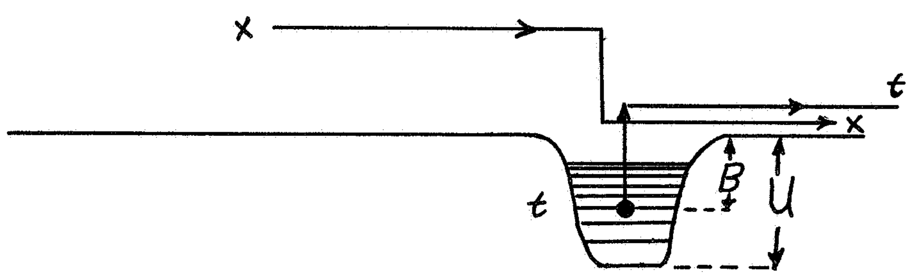

Figure 1

In high energy collisions the potential U which is about 60 MeV deep for low energy nucleons decreases and it may be set equal to zero for several hundred MeV incident energy. This simplifies the fact that the incoming particle will have to receive momentum from the nucleus when its energy increases from E to E + U inside the potential, and both particles give up momentum to the residual nucleus after leaving the potential

We will attempt a description by which the incoming particle x after entering the nucleus finds a particle, or particle cluster t and

*Professor of Physics, University of Virginia.

a residual nucleus N. The incident particle upon entering the nucleus will see an unbound t and N. Since we will have to satisfy energy and momentum at any instant the incident particle will loose momentum and energy as a consequence of it imparting energy to and breaking the binding of t and N. Also, longitudinal momentum will be imparted to the nucleus because one cannot satisfy the energy and momentum relations with 3 vectors of which two are equal in magnitude and oppositely oriented, as has to be the case for the momentum of t and the momentum of N in the nuclear frame. We have

$$\vec{p}_t + \vec{p}_N + \vec{p}_{in}' = \vec{p}_{in} \quad , \quad E_t + E_N + E_{in'} = E_{in}$$

and

$$\vec{p}_t + \vec{p}_N = 0 \quad , \quad \text{or} \quad E_t + E_N = M \quad \text{which is not generally true}$$

Therefore we set $\vec{p}_t' + \vec{p}_N' = 0$ in the nuclear frame S' which moves with the velocity Bc with respect to the laboratory frame after the impact with the incident particle. The motion of S' is a direct consequence of the unbinding of t and N. If t represents a nucleon then $\vec{p}_t'$ would be the Fermi momentum. S' provides the additional equation which is required to satisfy energy and momentum.*

The second part of the calculation describes the collision between the incident projectile and t in an ordinary two-body system.

These assumptions separate the problem into the following steps:

(1) A particle with mass m is incident on a nucleus of mass M. The three-momentum is $\vec{p}_p$, M is at rest.

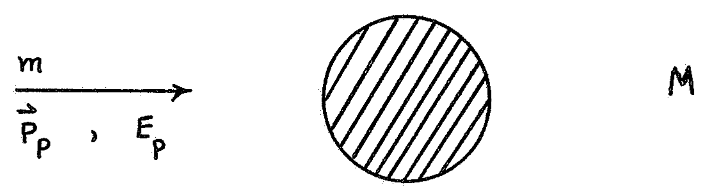

Figure 2

*The additional degree of freedom could have been obtained by setting $\vec{p}_N = \vec{p}_N'$. In that case only t would have taken up a momentum $\Delta\vec{p} = \vec{p}_t' - \vec{p}_t$.

The four-momentum is

$p_{in} = i\,\vec{p}_p + (E_p + M)$. Total energy is $E_{in} = E_p + M$

(2) The incident particle enters the nuclear field and consequently M splits up in N and t. The respective three-momenta are

$\vec{p}_o$ of the incident particle

$\vec{p}_t$ of particle t

$\vec{p}_N$ of the residual particle N

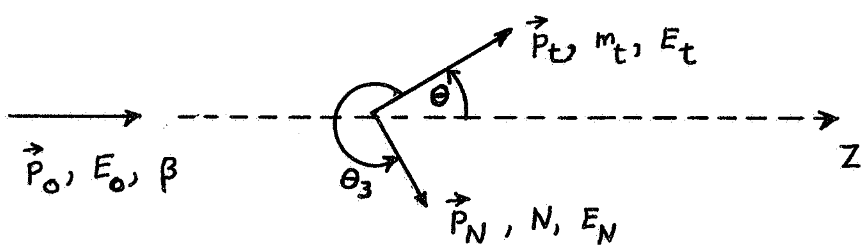

Figure 3

The four-momentum is $p_{in} = p_o + p_t + p_N$ (1)

The calculation will be carried out in a plane. A calculation in 3-dimensional space requires few changes. The generalization will only be necessary afterwards in the scattering of the incident particle and t.

In this process momentum had to be imparted to the nuclear frame. The respective quantities in the nuclear frame are presented in Fig. 4.

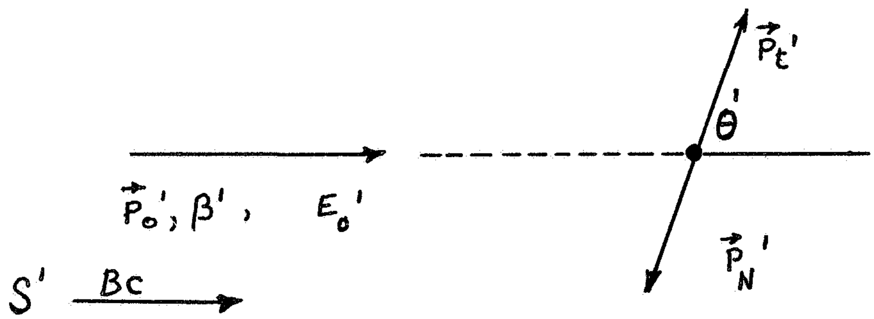

Figure 4

$\vec{p}_o{}'$ is the momentum of the incoming particle in S',

$\vec{p}_t{}'$ is the momentum of t in S' and

$\vec{p}_N{}'$ is the momentum of N in S'

$$\vec{p}_t{}' + \vec{p}_N{}' = 0. \tag{2}$$

The incident particle has a velocity $\beta' c$ and S' moves with respect to the laboratory with velocity Bc.

(3) The incident particle will then make a collision with t. This collision will be described as an ordinary two-body interaction.

II. Derivations

The first part of the calculation involves only the kinematics. We will use covariant expressions:

From step 2: $p_{in} = p_o + p_t + p_N$

$$\text{or} \quad (p_{in} - p_o)^2 = (p_t + p_N)^2 \tag{3}$$

This quantity is valid in any inertial frame. Therefore, we have

$$(p_t + p_N)^2 = (p_t' + p_N')^2 \tag{4}$$

or
$$(p_{in} - p_o)^2 = -(\vec{p}_t{}' + \vec{p}_N{}')^2 + (E_t{}' + E_N{}')^2 \tag{5}$$

According to (2) $\vec{p}_t{}' + \vec{p}_N{}' = 0$, and we can solve for E_o.

$$-\left(2p_t'^2 + m_t^2 + N^2 + 2\sqrt{(p_t'^2 + m_t^2)(p_t'^2 + N^2)}\right.$$

$$= p_p^2 + p_o^2 - 2p_pp_o - p_p^2 - m^2 - M^2 - p_o^2 - m^2 - 2E_pM + 2(E_p + M)\sqrt{p_o^2 + m^2} \tag{6}$$

or

$$p_t'^2 + \sqrt{(p_t'^2 + m_t^2)(p_t'^2 + N^2)} + \tfrac{1}{2}(N^2 - M^2 - 2m^2 + m_t^2) - E_pM = A$$

$$A = -(E_p + M)\sqrt{p_o^2 + m^2} + p_pp_o$$

or
$$A - p_p\sqrt{E_o^2 - m^2} = -E_{in}E_o \tag{7}$$

$$E_o = \frac{p_p\sqrt{A^2 - m^2(m^2 + 2E_pM + M^2)} - AE_{in}}{m^2 + 2E_pM + M^2} \tag{8}$$

Equation (6) defines A which is dependent on the momentum of t but not on its direction. Equation (8) presents the energy of the incident particle after the nucleus is dissociated. E_o is in the laboratory frame. The momentum is

$$p_o = \sqrt{E_o^2 - m^2}\ , \quad \beta = \frac{p_o}{E_o} \tag{9}$$

Presumably, one knows the momentum p_t' and p_N' in the nuclear frame S'. However, one does not know p_N and p_t in the laboratory frame. Consequently, we have to calculate p_o'.

The invariant quantity p_{in}^2 is in the laboratory frame,

$$p_{in}^2 = -\vec{p}_p^{\,2} + E_{in}^2 \tag{10}$$

In the nuclear frame S':

$$p_{in}^2 = -(\vec{p}_o' + \vec{p}_t' + p_N')^2 + (E_o' + E_t' + E_N')^2 \tag{11}$$

By solving for E_o', we find

$$E_o' = \frac{E_{in}^2 - p_p^2 - 2E_t'^2 - N^2 - m^2 + m_t^2 - 2E_t'E_N'}{2(E_t' + E_N')} \tag{12}$$

E_o' is the energy of the incident particle in the nuclear frame. Its momentum is:

$$p_o' = \sqrt{E_o'^2 - m^2} \tag{13}$$

and

$$\beta' = \frac{p_o'}{E_o'} \tag{14}$$

βc and $\beta' c$ are the velocities of the incident particle in the laboratory and in the nuclear frame respectively. Consequently, the nuclear frame moves with the velocity

$$B = \frac{\beta - \beta'}{1 - \beta\beta'} \tag{15}$$

$$\gamma = (1 - B^2)^{-1/2} \tag{16}$$

Now, p_t' and p_N' can be transformed into the laboratory system by the well known Lorentz transformation

$$(p_t)_z = p_t' \cos \Theta' + B\gamma\left(\frac{\gamma B p_t' \cos \Theta'}{1 + \gamma} + E_t'\right) \tag{17}$$

$$(p_t)_x = p_t' \sin \Theta' \tag{18}$$

Analogous equations transform $(p_n')_z$ and $(p_N')_x$

$$(p_N)_z = p_N' \cos (\pi - \Theta') + B\gamma\left(\frac{\gamma B p_N' \cos (\pi - \Theta')}{1 + \gamma} + E_N'\right) \tag{19}$$

$$(p_N)_x = p_N' \sin (\pi - \Theta') \tag{20}$$

This calculation does not put any constraints on m_t. In reactions of the kind $M(x, x\ t)N$, t could have rest mass zero or it could be a pion produced at the moment x enters the nucleus. M could be a nucleon or a composite nucleus.

III. Collision of x with t

This collision will be described in the center of mass system of x and t. The velocity of the center of mass is

$$\vec{\beta}_{cm} = \frac{\vec{p}_o + \vec{p}_t}{E_o + E_t} \tag{21}$$

The transformation of the momentum p for an elastic collision can be written symbolically

$$p_{final} = \left[\text{Lorentz } (\beta)\right]^{-1} \cdot \left[\text{rotation}\right] \cdot \left[\text{Lorentz } (\beta)\right] \cdot p \tag{22}$$

By making a separation between the dissociation of the nucleus and the two-body collision, we are able to consider other processes than elastic scattering. Such have to be described by appropriate transformations and other subroutines will have to be written.

The matrix for the Lorentz transformation is

$$\text{Lorentz } (\beta) = \begin{pmatrix} 1 + \beta_x^2 \kappa & \beta_x \beta_y \kappa & \beta_x \beta_z \kappa & i\beta_x \gamma \\ \beta_x \beta_y \kappa & 1 + \beta_y^2 \kappa & \beta_y \beta_z \kappa & i\beta_y \gamma \\ \beta_x \beta_z \kappa & \beta_y \beta_z \kappa & 1 + \beta_z^2 \kappa & i\beta_z \gamma \\ -i\beta_x \gamma & -i\beta_y \gamma & -i\beta_z \gamma & \gamma \end{pmatrix} \tag{23}$$

$\kappa = \gamma - 1$, and $\gamma = (1 - \beta^2)^{-1/2}$. We selected our coordinate system such that

$$\beta_y = \frac{p_{ty}}{E_o + E_t} = 0$$

The matrix for the momentum is

$$p = \begin{pmatrix} ip_x \\ 0 \\ ip_z \\ E \end{pmatrix} \tag{24}$$

The rotation matrix simplifies in the particular case for a vector in the XZ plane:

$$\begin{pmatrix} \cos\alpha\cos\epsilon & \cos\alpha\sin\epsilon & \sin\alpha\cos\epsilon & 0 \\ \cos\alpha\sin\epsilon & \cos\alpha\cos\epsilon & \sin\alpha\sin\epsilon & 0 \\ -\sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{25}$$

An additional simplification results if the rotation occurs only in the XZ plane.

$$\begin{pmatrix} \cos\alpha & 0 & \sin\alpha & 0 \\ 0 & \cos\alpha & 0 & 0 \\ -\sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (26)$$

In this particular $\alpha = \psi$ is the scattering angle. If the scattering would be out of the XZ plane, the scattering angle is given by

$$\cos\psi_{cm} = \sin(\alpha + \theta_{cm}) \sin\theta_{cm} \cos\epsilon + \cos(\alpha + \theta_{cm}) \cos\theta_{cm} \quad (27)$$

which reduces to

$$\cos\psi_{cm} = \cos\alpha \quad \text{for scattering in the XZ plane} \quad (28)$$

IV. Transformation of the Nuclear Momentum Distribution into the Experimentally Observed Cross Section

The momentum distribution in the nuclear frame is

$$\frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} = f(p_t') \quad (29)$$

The total of nucleons is:

$$N = \int \frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} p_t'^2 dp_t' \sin\theta' d\theta' d\phi' \quad (30)$$

In an (x, x t) experiment one will observe particles x and t in the elements of solid angle $d\Omega_1$ and $d\Omega_2$ in an energy interval $d(E_1 + E_2) = dE$. We have to relate the momentum distribution with the angular and energy distribution of the observed reaction products. The observation of the reaction products is made coplanar with the incident particle x.

The problem requires the construction of the Jacobian for the transformation:

$$\frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} p_t'^2 \, dp_t' \, d\Omega_N' \left(\frac{d\sigma}{d\Omega}\right)_{sc} d\Omega_{sc} \longrightarrow \frac{\partial^3 Z}{\partial E \partial \Omega_1 \partial \Omega_2} dE d\Omega_1 d\Omega_2$$

in which $\left(\frac{d\sigma}{d\Omega}\right)_{sc} d\Omega_{sc}$ represents the scattering cross section of particle x with particle t for scattering into the solid angle $d\Omega_{sc}$.

The first part of the calculation will be the transformation into the cm-system of incoming particle x and particle t.

$$\mathcal{N} = \frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} p_t'^2 \, d\varphi \begin{vmatrix} \dfrac{\partial p_t'}{\partial p_{cm}} & \dfrac{\partial p_t'}{\partial (\cos X)} \\ \dfrac{\partial \cos \theta'}{\partial p_{cm}} & \dfrac{\partial \cos \theta'}{\partial (\cos X)} \end{vmatrix} d(\cos X) dp_{cm} \tag{31}$$

p_{cm} is the cm momentum.

X is the angle with respect to the z-axis in the cm system.

The second part of the calculation is the collision of x and t. The cross section is

$$\frac{d\sigma(\psi)}{d\Omega_{cm}} d(\cos \psi) d\varphi$$

ψ is the scattering angle in the cm system. Actually the scattering occurs from the angle X into the angle $X + \psi$. The number of collisions per incoming particle is,

$$Z = \mathcal{N} \cdot \frac{d\sigma}{d\Omega_{cm}} \cdot \sin(X + \psi) \, d\psi \, d\varphi \tag{32}$$

$$Z = \frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} p_t'^2 \frac{d\sigma}{d\Omega_{cm}} d\varphi \begin{vmatrix} \frac{\partial p_t'}{\partial p_{cm}} & \frac{\partial p_t'}{\partial (\cos X)} \\ \frac{\partial (\cos \theta')}{\partial p_{cm}} & \frac{\partial (\cos \theta')}{\partial (\cos X)} \end{vmatrix} \sin(X + \psi)\, d\psi\, d(\cos X)\, dp_{cm} d\varphi \tag{33}$$

This expression has to be transformed in the laboratory system with the variables θ_1 , θ_2 and E.

$$Z = \frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} \cdot \frac{d\sigma}{d\Omega_{cm}} \cdot (d\varphi)^2\, p_t'^2\, \frac{\partial(p_t', \cos \theta')}{\partial(p_{cm}, \cos X)} \cdot \frac{\partial(p_{cm}, \cos X, \psi)}{\partial(\cos \theta_1, \cos \theta_2, E)} \cdot$$

$$\sin(X + \psi)\, d(\cos \theta_1)\, d(\cos \theta_2)\, dE \tag{34}$$

The multiplication of the determinants can be carried out easily. One obtains finally:

$$Z = \frac{\partial^2 n}{\partial p_t' \partial \Omega_N'} \frac{d\sigma}{d\Omega_{cm}} p_t'^2 \sin (X + \psi)(d\varphi)^2 d(\cos \theta_1)\, d(\cos \theta_2)\, dE\, J \tag{35}$$

$$J = \begin{vmatrix} \frac{\partial p_t'}{\partial (\cos \theta_1)} & \frac{\partial p_t'}{\partial (\cos \theta_2)} & \frac{\partial p_t'}{\partial E} \\ \frac{\partial (\cos \theta')}{\partial (\cos \theta_1)} & \frac{\partial (\cos \theta')}{\partial (\cos \theta_2)} & \frac{\partial (\cos \theta')}{\partial E} \\ \frac{\partial \psi}{\partial (\cos \theta_1)} & \frac{\partial \psi}{\partial (\cos \theta_2)} & \frac{\partial \psi}{\partial E} \end{vmatrix} \tag{36}$$

V. Discussion

This calculation is based on the use of a particular model

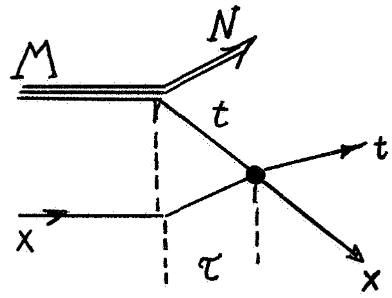

Figure 5

as presented in Figure 5; M breaks up when x enters the nucleus. At any instant, all particles are on the energy shell. x gives up energy and momentum upon entering the interval τ.

A quantum mechanical calculation would consider t off the energy shell in the interval τ to place t back on the energy shell after the collision. The transition amplitude is:

$$T_{fi} = \langle \Phi^{(-)}(\vec{K}_f) \, |V| \, \Phi^{(+)}(\vec{K}_i) \rangle \tag{37}$$

In Born approximation we would have:

$$\Phi^{(+)}(\vec{K}_i) = {}_M(\vec{r}_1 \ldots \vec{r}_A) \; e^{i\vec{k}_i \cdot \vec{r}_i} \tag{38}$$

$$\Phi^{(-)}(\vec{K}_f) = \Psi_N(\vec{r}_1 \ldots \vec{r}_{A-1}) e^{-i\vec{k}_{fi} \cdot \vec{r}_{fi}} \cdot e^{-i\vec{k}_{f2} \cdot \vec{r}_{f2}}$$

$$\Psi_M(\vec{r}_1 \ldots \vec{r}_A) = \Psi_N(\vec{r}_1 \ldots \vec{r}_{A-1}) \; \Psi(\vec{r}_t)$$

$\Psi(\vec{r}_t)$ is the wave function of the particle t.

The cross section for the reaction will be,

$$\sigma = \frac{2\pi}{\hbar} \sum_{av} |T_{fi}|^2 \rho(\vec{K}_i) \; \delta(\vec{p}_{in} - \vec{p}_1 - \vec{p}_2 - \vec{p}_N) \tag{39}$$

The δ function requires energy and momentum conservation. The interaction Hamiltonian V contains all the information about the interaction of t and x.

The kinematic calculation replaces this interaction by an experimental scattering cross section between t and x.

The summation over all different possible scatterings have to be carried out, e.g., the diagram of Fig. 6 contributes in the same way as the diagram of Fig. 5.

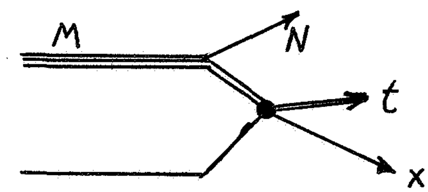

Figure 6

The cross section of x elastically scattered from t which is now a composite particle has to be included.

The calculation is not limited to scatterings of an incoming particle with a nucleon or nuclear cluster. Instead t may well be generated as a pion or a photon

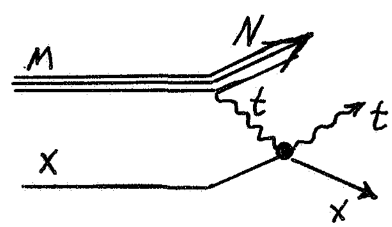

Figure 7

Actually the diagram of Fig. 7 is not entirely correct for the description of the process. Since x gives up the energy for the production the diagram is as follows:

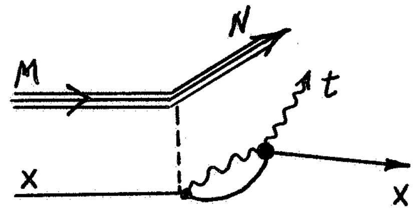

Figure 8

A small change in the program could be made in order to have N scattered by t as represented in Fig. 9. A statement which interchanges N and t before the scattering would affect this.

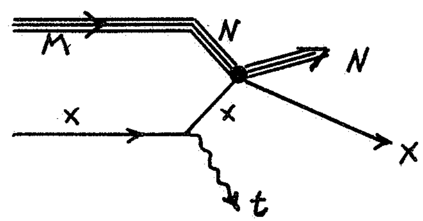

Figure 9

The spectrum for t in any of the processes described above could be fed to the computer. If t is a nucleon in the nucleus one could require p_t' to be the momentum distribution of a particle in a nuclear shell. If t is a photon one should assume a bremsstrahlung spectrum.

Finally one could read in a table of scattering cross section and an additional very simple subroutine could calculate the final result for the reaction cross section.

VI. Computer Program

Two programs are included in the appendix. The first program calculates tables for the kinematic quantities for a M(x, x t)N reaction.

That calculation is separated in two parts:

1. The separation of the nucleus M into t and N.

2. The Lorentz transformation into a cm frame of t and x, and a rotation of the four-momenta corresponding to a scattering by an angle ψ. The print-out is in the main program.

The other program calculates in addition to the kinematic quantities the Jacobian for the transformation from the nuclear system to the laboratory system.

This calculation is separated in the following parts:

1. The main program which contains the read and write statements, the increments for the nuclear momentum and the angle of the nuclear momentum with respect to the incoming direction. It also contains the increments for the calculation of the Jacobian.

2. A subroutine which treats the transformation of the lab-system into the nuclear system. It calculates the formulas 7 through 12.

3. A subroutine which treats the dissociation of the nuclear system. It calculates the formulas up to 20.

4. A subroutine which calculates the scattering of x and t.

5. A subroutine which multiplies 4 dimensional complex matrixes.

The calculation of the Jacobian is effected by carrying out small variations in p_t', Θ' and ψ. One calculates then the variations in Θ_1, Θ_2 and E.

VII. Final Results

The calculations on the D(p, 2p)n reaction have been carried out as an example. The increments of p_t', Θ' and ψ were varied in the range from 0.5 to 2 MeV and 0.5° to 2° respectively. The variations in J were mostly within the range of the required precision of about 2%. However, large variations were encountered for $\Theta = 0^\circ$ and $\Theta = 180^\circ$. For this reason the values for J at these angles were obtained by extrapolation. Figure 10 shows the symmetric angle of scattering vs the nuclear momentum. Figure 11 presents

$$\frac{\partial(\cos\Theta_1, \cos\Theta_2, E)}{\partial(\cos\Theta', \cos(X+\psi), p_t')}$$

as a function of Θ'.

The momentum p_t' is a parameter and the scattering angle $\psi = 90^{\circ}$. From this figure the determinant for $\Theta' = 0^{\circ}$ and $\theta' = 180^{\circ}$ is determined. Fig. 12 and Fig. 13 present the determinant as a function of p_t'. Fig. 14 shows the value of

$$\frac{d\sigma(90^{\circ})}{d\Omega_{cm}}$$

as a function of E. Finally Fig. 15 presents the complete expression,

$$T = \frac{\partial(\cos\Theta_1' \cos(X+\psi), p_t')}{\partial(\cos\Theta_1, \cos\Theta_2, E)} p_t'^2 \frac{d\sigma}{d\Omega_{cm}}$$

for $\psi = 90^{\circ}$, and $\Theta' = 0^{\circ}$ and $\Theta' = 180^{\circ}$. To obtain the experimentally observable distribution one has to assume a momentum distribution for the proton inside the deuteron

$$\frac{\partial^2 n}{\partial p_t' \partial \Omega'_N}$$

which has to be multiplied with T,

$$Z_o d\Omega_1\, d\Omega_2\, dE = \frac{\partial^2 n}{\partial p_t' \partial \Omega_N} T\, d(\cos\Theta_1)\, d\varphi\, d(\cos\Theta_2) d\varphi\, dE$$

This calculation neglects all final state interactions and it neglects the absorption of x and of t on its passage through the nucleus. Known reaction cross sections make possible the application of a correction for the absorption. However, the final state interaction cannot be corrected for in a simple manner.

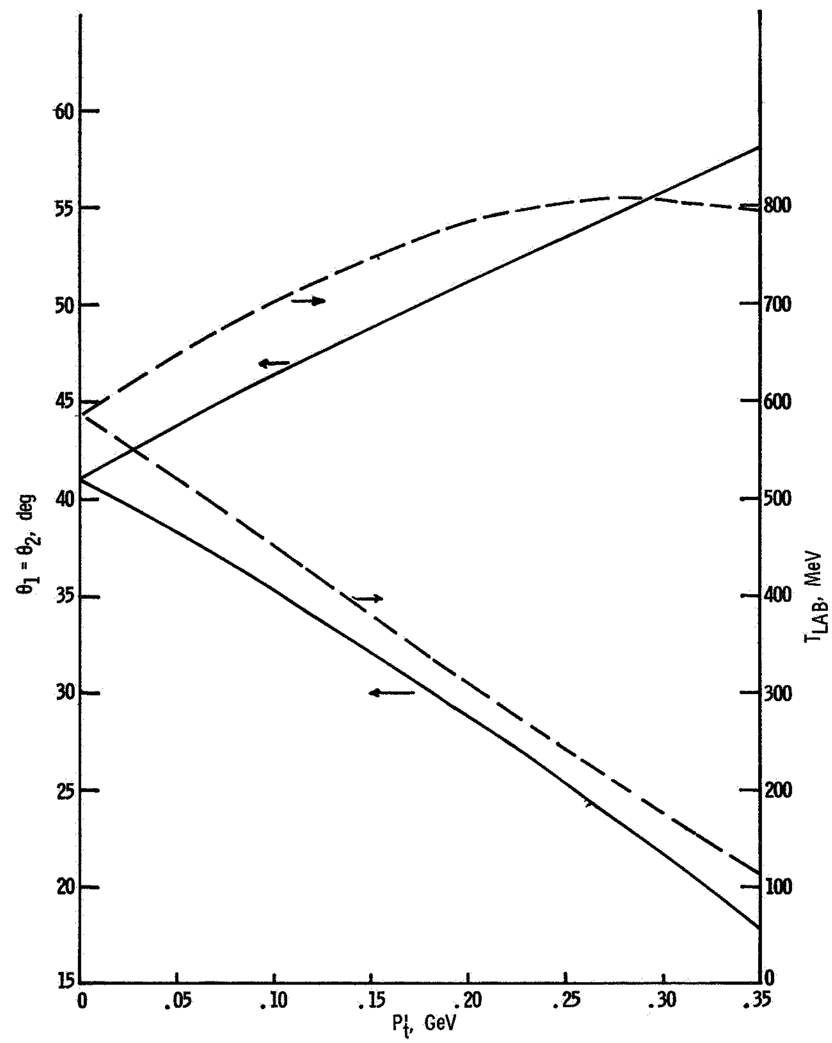

Figure 10. - Solid curve is the symmetric scattering angle $\theta_1 = \theta_2$ vs nuclear momentum $p_t^{\prime}$, dashed curve is the symmetric scattering angle vs the p-p energy in a system where one proton is at rest. The reaction is D(p, 2p)n, Ep = 590 MeV.

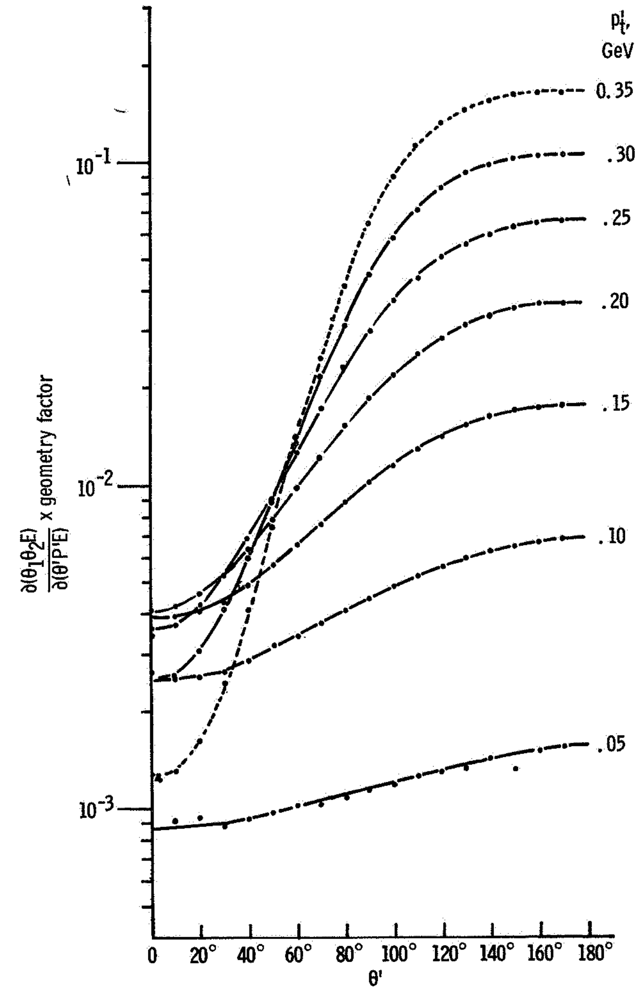

Figure 11. - Transformation Jacobian vs θ' for the reaction D(p, 2p)n; Ep = 590 MeV, ψsc = 90°.

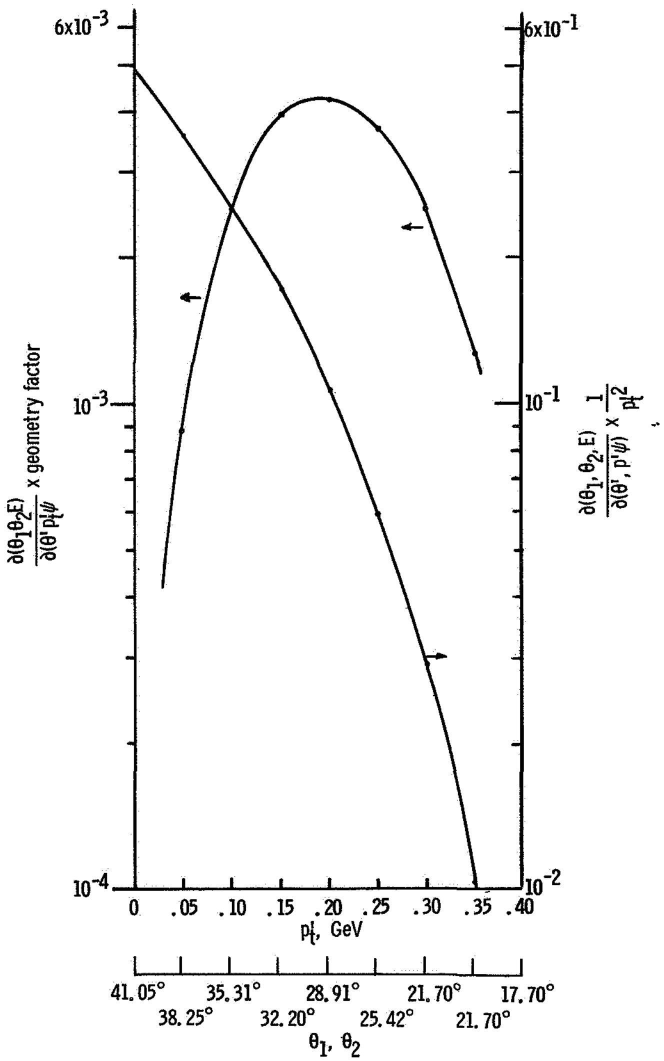

Figure 12. - Transformation Jacobian vs p_t' for the reaction D(p, 2p)n; Ep = 590 MeV, ψsc = 90°, θ = 0°. p_t' is the momentum of the bound proton in the nuclear frame.

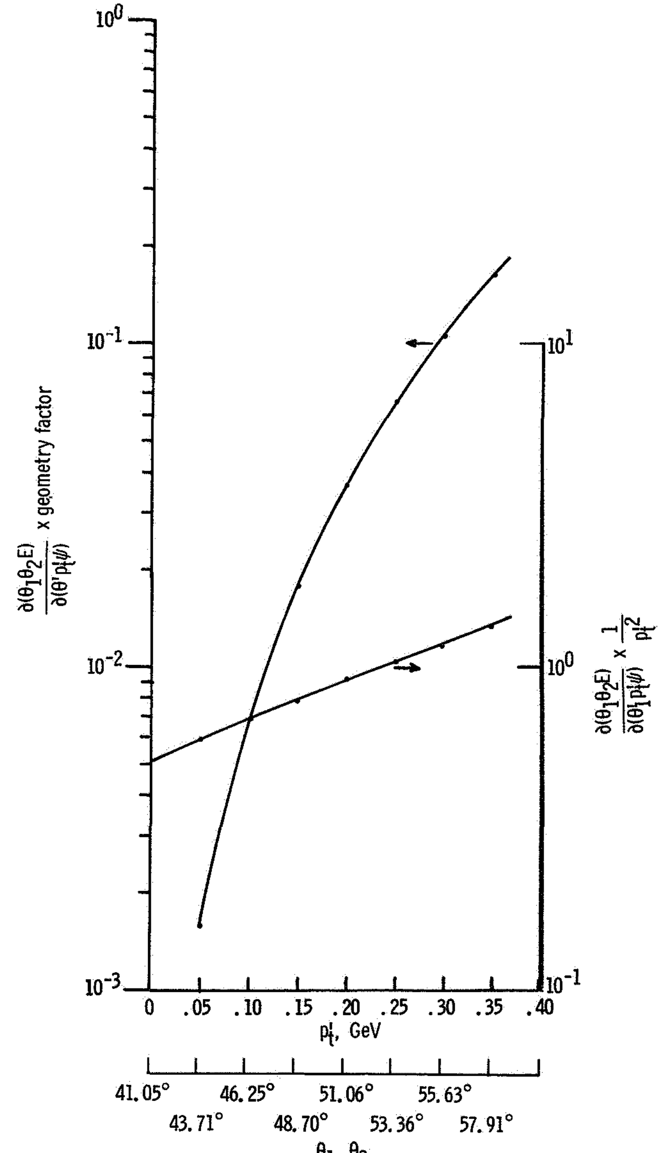

Figure 13. - Transformation Jacobian vs p_t' for the reaction D(p, 2p)n; Ep = 590 ψsc = 90°, θ' = 180°. p_t' is the momentum of the bound proton in the nuclear frame.

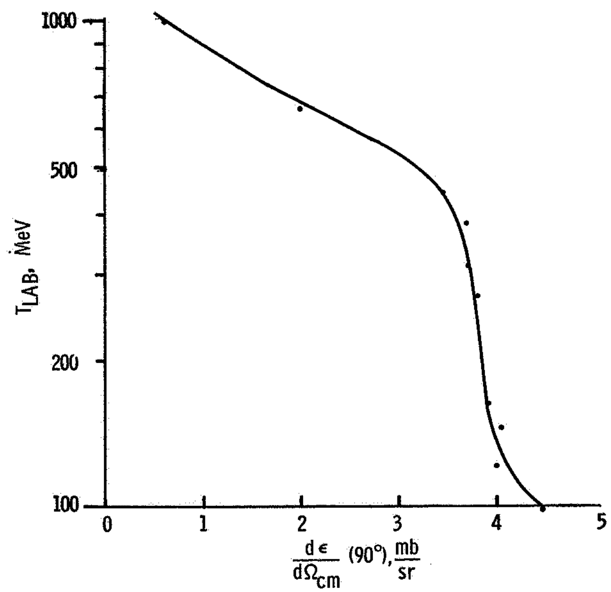

Figure 14. - Laboratory energy, T vs $d\sigma/d\Omega$ at 90° in the center of mass system for p-p scattering.

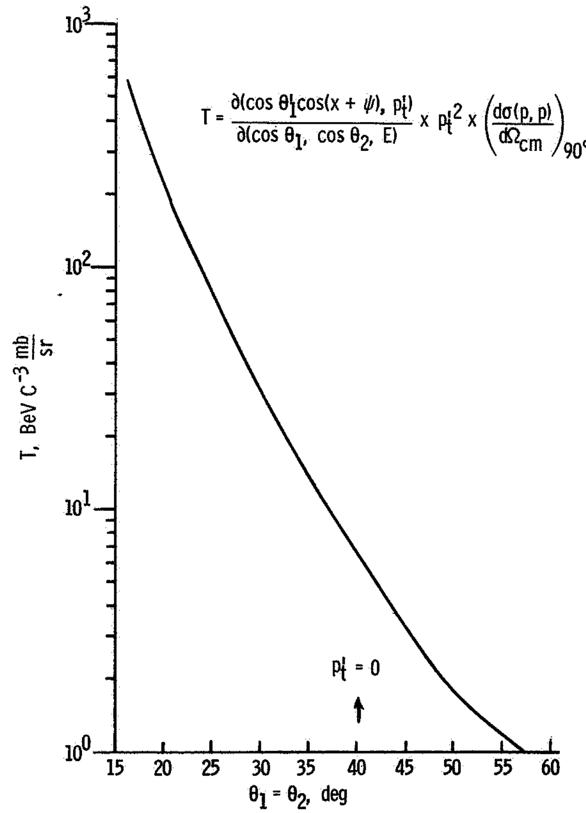

Figure 15. - Transformation Jacobian vs θ_1, θ_2 for the reaction D(p, 2p)n; Ep = 590 MeV.

KINEMATICS M(X,Xt)

```
C      ALL ENERGIES AND MASSES ARE TO BE READ IN IN GEV
C      INCIDENT PROTON KINETIC ENERGY  ENIN
C      A CLUSTER CAN BE ANY SUBUNIT OF THE TARGET NUCLEUS
C      TARGET CLUSTER  MOMENTUM PT'  PTGT
C      INCIDENT PROTON ENERGY EP  ENINM
C      INCIDENT TOTAL  ENERGY EIN ENINT
C      INCIDENT PROTON MOMENTUM PP  PMOMI
C      TARGET CLUSTER ENERGY ET'  ENTGT
C      RECOILING RESIDUAL NUCLEUS ENERGY EN'  ENRC
C      INCIDENT PROTON MOMENTUM AFTER BREAK UP OF NUCL. PO  PBK
C      INCIDENT PROTON ENERGY AFTER BREAK UP OF NUCL.IN MOV FRAME  ENMOV
C      INCIDENT PROT. MOM. AFTER BREAK UP OF NUCL. IN MOV FRAME PMOV
C      BETA IN LAB FRAME BETI
C      BETA' IN MOVING FRAME BETM
C      BETA OF MOVING FRAME BETFR
C      THETA IN MOVING FRAME  THTG
C      GAMA OF MOVING FRAME  GAM
C      MOMENTUM OF TARGET CLUSTER BEFORE COLLISON PT  PT
C      THETA IN LAB FRAME THT
C      ENERGY OF RECOIL IN LAB FRAME EN  ENN
C      ENKTO,DENKT,ENKTM ARE THE MIN, INCREMENT, AND MAX NUCL MOMENTA
       IMPLICIT REAL*4(M)
       COMMON SCAN,PBK,PTZ,PTX,ENK,ENT,FINEN,FITEN,FANG,FANG2,FINMX,FINMZ
      1,PCMEN ,MASIN
     1 READ(5,10)ENIN,MASIN,MASTG,MASR ,MASCL
    10 FORMAT(5F10.5)
       READ(5,11)ENKTO,DENKT,ENKTM
       READ(5,11)THTGO,DTHTG,THTGM
       READ(5,11)SCANO,DSCAN,SCANM
    11 FORMAT(3F10.5)
       WRITE(6,12)ENIN,MASIN,MASTG,MASCL,MASR
    12 FORMAT(1H1,'KINEMATICS FOR (X,XY)  '/18H INCIDENT ENERGY =
      1F10.5/ 16H INCIDENT MASS =  F10.5/ 14H TARGET MASS =  F10.5
      2/ 15H MASS CLUSTER =  F10.5  /24H MASS RESIDUAL NUCLEUS =  F10.5)
       ENINM=ENIN+MASIN
       ENINT=ENINM+MASTG
       PMOMI=SQRT(ENIN**2+2.0*MASIN*ENIN)
       PTGT=ENKTO
       PTGT=PTGT-DENKT
    15 PTGT=PTGT+DENKT
       ENTGT=SQRT(PTGT**2+MASCL**2)
       ENRC=SQRT(PTGT**2+MASR**2)
       VAR1=MASIN**2+MASTG**2+2.0*MASTG*ENINM
       ADA=PTGT**2+0.5*(MASR**2-MASTG**2-2.0*MASIN**2+MASCL**2)
      1-ENINM*MASTG+ENTGT*ENRC
       ENK=(PMOMI*SQRT(ADA**2-VAR1*MASIN**2)-ADA*ENINT)/VAR1
       PBK=SQRT(ENK**2-MASIN**2)
       ENMOV=(ENINT**2-PMOMI**2- 2.0*ENTGT**2-MASR**2-2.0*ENTGT*ENRC+
      1MASCL **2-MASIN**2)/(2.0*(ENTGT+ENRC))
```

```
      PMOV=SQRT(ENMOV**2-MASIN**2)
      BETI=PBK/ENK
      BETM=PMOV/ENMOV
      BETFR=(BETI-BETM)/(1.0-BETI*BETM)
      GAM=SQRT(1.0/(1.0-BETFR**2))
      PX=PMOV+BETFR*GAM*(GAM*BETFR*PMOV/(1.0+GAM)+ENMOV)
      EX=SQRT(PX**2+MASIN**2)
      ENRCK=(ENRC-MASR)*1000.0
      ENMOK=(ENMOV-MASIN)*1000.0
      EXK=(EX-MASIN)*1000.0
      THTG=THTGO-DTHTG
   19 THTG=(THTG+DTHTG)*3.141593/180.0
      PTZ=PTGT*COS(THTG)+BETFR*GAM*(GAM*BETFR*PTGT*COS(THTG)/(1.0+GAM)+
     1ENTGT)
      PTX=PTGT*SIN(THTG)
      PT=SQRT(PTZ**2+PTX**2)
      IF(PTX.EQ.0.0.AND.PTZ.EQ.0.0) PTZ=0.00001
      THT=ATAN2(PTX,PTZ)
      ENT=SQRT(PT**2+MASCL**2)
      PNX=PTGT*SIN(3.141593+THTG)
      PNZ=PTGT*COS(3.141593+THTG)+BETFR*GAM*(GAM*BETFR*PTGT*
     1COS(3.141593+THTG)/(1.0+GAM)+ENRC)
      IF(PNX.EQ.0.0.AND.PNZ.EQ.0.0)PNZ=0.00001
      THN=ATAN2(PNX,PNZ)
      PN=SQRT(PNZ**2+PNX**2)
      ENN=SQRT(PN**2+MASR**2)
      ET=ENN+ENT+ENK
      THT=180.0*THT/3.141593
      THTG=180.0*THTG/3.141593
      THN=180.0*THN/3.141593
      ENTK=(ENT-MASCL)*1000.0
      ENKK=(ENK-MASIN)*1000.0
      ENTGK=(ENTGT-MASCL)*1000.0
      ENNK=(ENN-MASR)*1000.0
      WRITE(6,189)
  189 FORMAT(1HO' CLUST K E  KIN E REC   INC E MOV  BETA IN   BETA MOV
     1BETA FR     P COLL EN     ')
      WRITE(6,18)ENTGK,ENRCK,ENMOK,BETI,BETM,BETFR,EXK
   18 FORMAT(7(F10.4,1X))
      WRITE(6,190)
  190 FORMAT(1HO'EN TGT     P COLL EN  ANGLE   TGT CL EN   REC EN    TOT E
     1N     THETA CL  ANGLE REC ')
      WRITE(6,21)ENTGK,ENKK,THTG,ENTK,ENNK,ET,THT,THN
   21 FORMAT(8(F9.4,1X))
      WRITE(6,191)
  191 FORMAT(1HO' INC MOM  CLST MOM  BR INC M  MOV INC M  MOV INC   TGT M
     1OM   TGT MOMX  REC MOM   ')
      WRITE(6,21)PMOMI,PTGT,PBK,PMOV,PX,PT,PTX,PN
      WRITE(6,30)
```

```
30 FORMAT(1HO' CM ANGLE  L.ANGLE 1 ENER 1     L.ANGLE 2 ENER 2      TOT
  1EN     MOM TR SQ   C CM EN   P LAB EN          ')
   SCAN=SCANO-DSCAN
20 SCAN=(SCAN+DSCAN)*3.141593/180.0
   CALL SCATT
   TOT=FINEN+FITEN+ENN
   TEE=-(PMOMI**2+FINMX**2+FINMZ**2-2.0*PMOMI*FINMZ)+(ENINM-FINEN)**2
   SCAN=180.0*SCAN/3.141593
   FANG=180.0*FANG/3.141593
   FANG2=180.0*FANG2/3.141593
   FINEN=(FINEN-MASIN)*1000.0
   FITEN=(FITEN-MASCL)*1000.0
   PCMEK=(PCMEN-MASIN-MASCL)*1000.0
   IF(MASCL.NE.0.0) GO TO 140
   PLABE=((PCMEN**2-MASIN**2)/(2.0*MASIN))*1000.0
   GO TO 141
140 CONTINUE
   PLABE=((PCMEN**2-MASIN**2-MASCL**2)/(2.0*MASCL)-MASIN)*1000.0
141 CONTINUE
   WRITE(6,31)SCAN,FANG,FINEN,FANG2,FITEN,TOT,TEE ,PCMEK,PLABE
31 FORMAT(F7.2,3X,8(F9.4,1X))
   IF(SCAN.LE.SCANM)GO TO 20
   SCAN=SCANO
   IF(PTGT.EQ.0.0)GO TO 15
   IF(THTG.LE.THTGM)GO TO 19
   THTG=THTGO
   IF(PTGT.LE.ENKTM)GO TO 15
100 CONTINUE
   GO TO 1
   END
```

```
      SUBROUTINE SCATT
      IMPLICIT COMPLEX*8(A) ,REAL*4(M)
      COMMON SCAN,PBK,PTZ,PTX,ENK,ENT,FINEN,FITEN,FANG,FANG2,FINMX,FINMZ
     1,PCMEN ,MASIN
      DIMENSION APK(4,4),APT(4,4),AA(4,4),APKCM(4,4),APTCM(4,4),
     1AROT(4,4),AFINM(4,4),AFINT(4,4), APTSC(4,4),APKSC(4,4)
      BETAX=-PTX/(ENK+ENT)
      BETAZ=-(PBK+PTZ)/(ENK+ENT)
      BETSQ=BETAZ**2+BETAX**2
      APK(1,1)=(0.0,0.0)
      APK(2,1)=(0.0,0.0)
      APK(3,1)= CMPLX(0.0,PBK)
      APK(4,1)=CMPLX(ENK,0.0)
      APT(1,1)=CMPLX(0.0,PTX)
      APT(2,1)=(0.0,0.0)
      APT(3,1)=CMPLX(0.0,PTZ)
      APT(4,1)=CMPLX(ENT,0.0)
      DO 5 IP=1,4
      DO 5 JP=2,4
      APK(IP,JP)=(0.0,0.0)
    5 APT(IP,JP)=(0.0,0.0)
      GAMM=1.0/SQRT(1.0-BETSQ)
      HAMM=GAMM-1.0
      INDEX=0
   10 AA(1,1)=CMPLX(1.0+HAMM*BETAX**2/BETSQ,0.0)
      AA(1,2)=(0.0,0.0)
      AA(1,3)=CMPLX(BETAX*BETAZ*HAMM/BETSQ,0.0)
      AA(1,4)=CMPLX(0.0,BETAX*GAMM)
      AA(2,1)=(0.0,0.0)
      AA(2,2)=(1.0,0.0)
      AA(2,3)=(0.0,0.0)
      AA(2,4)=(0.0,0.0)
      AA(3,1)=AA(1,3)
      AA(3,2)=(0.0,0.0)
      AA(3,3)=CMPLX(1.0+HAMM*BETAZ**2/BETSQ,0.0)
      AA(3,4)=CMPLX(0.0,BETAZ*GAMM)
      AA(4,1)=CMPLX(0.0,-BETAX*GAMM)
      AA(4,2)=(0.0,0.0)
      AA(4,3)=CMPLX(0.0,-BETAZ*GAMM)
      AA(4,4)=CMPLX(GAMM,0.0)
      IF(INDEX.EQ.1) GO TO 20
      CALL MMULT(AA,APK,APKCM)
      CALL MMULT(AA,APT,APTCM)
      PCMEN=REAL(APKCM(4,1))+REAL(APTCM(4,1))
      AROT(1,1)=CMPLX(COS(SCAN),0.0)
      AROT(1,2)=(0.0,0.0)
      AROT(1,3)=CMPLX(SIN(SCAN),0.0)
      AROT(1,4)=(0.0,0.0)
      AROT(2,1)=(0.0,0.0)
```

```
      AROT(2,2)=(1.0,0.0)
      AROT(2,3)=(0.0,0.0)
      AROT(2,4)=(0.0,0.0)
      AROT(3,1)=CMPLX(-SIN(SCAN),0.0)
      AROT(3,2)=(0.0,0.0)
      AROT(3,3)=CMPLX(COS(SCAN),0.0)
      AROT(3,4)=(0.0,0.0)
      AROT(4,1)=(0.0,0.0)
      AROT(4,2)=(0.0,0.0)
      AROT(4,3)=(0.0,0.0)
      AROT(4,4)=(1.0,0.0)
      CALL MMULT(AROT,APKCM,APKSC)
      CALL MMULT(AROT,APTCM,APTSC)
      BETAX=-BETAX
      BETAZ=-BETAZ
      INDEX=1
      GO TO 10
   20 CONTINUE
      CALL MMULT(AA,APKSC,AFINM)
      CALL MMULT(AA,APTSC,AFINT)
      FINMX=AIMAG(AFINM(1,1))
      FINMZ=AIMAG(AFINM(3,1))
      FINEN=REAL(AFINM(4,1))
      FINTX=AIMAG(AFINT(1,1))
      FINTZ=AIMAG(AFINT(3,1))
      FITEN=REAL(AFINT(4,1))
      FANG=ATAN2(FINMX,FINMZ)
      FANG2=ATAN2(FINTX,FINTZ)
      RETURN
      END
```

```
      SUBROUTINE MMULT (AA,AB,AC)
C PROGRAM MULTIPLIES THE 4-DIM. MATRICES AA(IA,JA) AND AB(IB,JB)
      IMPLICIT COMPLEX*8(A)
      DIMENSION AA(4,4),AB(4,4),AC(4,4)
      DO 10 IA=1,4
      DO 10 JB=1,4
      AC(IA,JB)=AA(IA,1)*AB(1,JB)+AA(IA,2)*AB(2,JB)+AA(IA,3)*AB(3,JB)+
     1AA(IA,4)*AB(4,JB)
   10 CONTINUE
      RETURN
      END
```

X,Xt KINEMATICS AND TRANSF. JACOBIAN

```
C       ALL ENERGIES AND MASSES *ARE TO BE READ IN IN GEV
C       INCIDENT PROTON KINETIC ENERGY  ENIN
C       A CLUSTER CAN BE ANY SUBUNIT OF THE TARGET NUCLEUS
C       TARGET CLUSTER  MOMENTUM PT'  PTGT
C       INCIDENT PROTON ENERGY EP  ENINM
C       INCIDENT TOTAL  ENERGY EIN ENINT
C       INCIDENT PROTON MOMENTUM PP  PMOMI
C       TARGET CLUSTER ENERGY ET'  ENTGT
C       RECOILING RESIDUAL NUCLEUS ENERGY EN'  ENRC
C       INCIDENT PROTON MOMENTUM AFTER BREAK UP OF NUCL. PO  PBK
C       INCIDENT PROTON ENERGY AFTER BREAK UP OF NUCL.IN MOV FRAME  ENMOV
C       INCIDENT PROT. MOM. AFTER BREAK UP OF NUCL. IN MOV FRAME PMOV
C       BETA IN LAB FRAME BETI
C       BETA' IN MOVING FRAME BETM
C       BETA OF MOVING FRAME BETFR
C       THETA IN MOVING FRAME  THTG
C       GAMA OF MOVING FRAME  GAM
C       MOMENTUM OF TARGET CLUSTER BEFORE COLLISON PT  PT
C       THETA IN LAB FRAME THT
C       ENERGY OF RECOIL IN LAB FRAME EN  ENN
        IMPLICIT REAL*4(M)
        COMMON MASIN,MASTG,MASCL,MASR,ENINM,ENINT,ENMOV,PMOV,BETI,BETM,
       1PT,THT,PNX,PNZ,THN,PN,ENN,ET,PX,EX,FINMX,FINMZ,PMOMI
       2,VREL
      1 READ(5,10)ENIN,MASIN,MASTG,MASR ,MASCL
     10 FORMAT(5F10.5)
        READ(5,11)ENKTO,DENKT,ENKTM
        READ(5,11)THTGO,DTHTG,THTGM
        READ(5,11)SCANO,DSCAN,SCANM
        READ(5,11)PSMAL,TSMAL,SSMAL
     11 FORMAT(3F10.5)
        MASR=MASR*MASIN  /1.0078253   * MASR was read in AMU.
        THTGO=THTGO*0.0174532925
        DTHTG=DTHTG*0.0174532925
        THTGM=THTGM*0.0174532925
        SCANO=SCANO*0.0174532925
        DSCAN=DSCAN*0.0174532925
        SCANM=SCANM*0.0174532925
        TSMAL=TSMAL*0.0174532925
        SSMAL=SSMAL*0.0174532925
        WRITE(6,12)ENIN,MASIN,MASTG,MASCL,MASR
     12 FORMAT(1H1,'KINEMATICS FOR (X,XY)  '/18H INCIDENT ENERGY =
       1F10.5/ 16H INCIDENT MASS =  F10.5/ 14H TARGET MASS =  F10.5
       2/ 15H MASS CLUSTER =  F10.5  /24H MASS RESIDUAL NUCLEUS =  F10.5)
        ENINM=ENIN+MASIN
        ENINT=ENINM+MASTG
        PMOMI=SQRT(ENIN**2+2.0*MASIN*ENIN)
        PTGT=ENKTO
        PTGT=PTGT-DENKT
```

```
 15 PTGT=PTGT+DENKT
    CALL NUCSYS(PTGT,ENK,PBK,BETFR,ENTGT,ENRC)
    THTG=THTGO-DTHTG
 19 THTG=THTG+DTHTG
    CALL DISSOC(PTGT,THTG,BETFR,ENTGT,ENRC,PTZ,PTX,ENT)
    ENTK=(ENT-MASCL)*1000.0
    ENKK=(ENK-MASIN)*1000.0
    ENTGK=(ENTGT-MASCL)*1000.0
    ENNK=(ENN-MASR)*1000.0
    ENRCK=(ENRC-MASR)*1000.0
    ENMOK=(ENMOV-MASIN)*1000.0
    EXK=(EX-MASIN)*1000.0
    THTGA=THTG*57.2957795
    THNA=THN*57.29577951
    THTA=THT*57.29577951
    ET=ENN+ENT+ENK
    WRITE(6,110)PMOMI,PTGT,ENTGK,THTGA,PN,ENNK,THNA,BETFR,ET
110 FORMAT(1HO'INCOMING MOMENTUM ='F10.4  /
   119H CLUSTER MOMENTUM=  F10.4, 17H CLUSTER ENERGY=   F7.2, 8H ANGLE
   2=   F6.1  /18H RECOIL MOMENTUM=   F10.4,   16H RECOIL ENERGY=
   3F7.2,   8H ANGLE =    F6.1/   18H NUCLEAR SYS BETA=   F10.4,
   415H TOTAL ENERGY=   F10.5,   4H GEV   )
    WRITE(6,191)
191 FORMAT(1HO' INC MOM  CLST MOM  BR INC M  MOV INC M  MOV INC  TGT M
   1OM  TGT MOMX  REC MOM   ')
    WRITE(6,21)PMOMI,PTGT,PBK,PMOV,PX,PT,PTX,PN
 21 FORMAT(8(F9.4,1X))
    WRITE(6,35)
 35 FORMAT(1H ' TOT EN   MOM TR SQ  C.CM EN  CM REL VEL  COL.CM AN
   1 XY LAB ENERG  ')
    IK=0
    SCAN=SCANO-DSCAN
 20 SCAN=SCAN+DSCAN
    CALL SCATT(PTX,PBK,ENK,ENT,PTZ,SCAN,FINEN,FITEN,FANG,FANG2,
   1PCMEN,CMAN)
    TOT=FINEN+FITEN+ENN
    TEE=-(PMOMI**2+FINMX**2+FINMZ**2-2.0*PMOMI*FINMZ)+(ENINM-FINEN)**2
    PCMEK=(PCMEN-MASIN-MASCL)*1000.0
    VEREL=VREL
    IF(MASCL.NE.0.0) GO TO 140
    PLABE=((PCMEN**2-MASIN**2-MASCL**2)/(2.0*MASIN))*1000.0
    GO TO 141
140 CONTINUE
    PLABE=((PCMEN**2-MASIN**2-MASCL**2)/(2.0*MASCL)-MASIN)*1000.0
141 CONTINUE
    FINEK=1000.0*(FINEN-MASIN)
    FITEK=1000.0*(FITEN-MASCL)
    CMAA=CMAN*57.29577951
    FNGA=FANG*57.29577951
```

```
      FAA2A=FANG2*57.29577951
      SCNA=SCAN*57.29577951
      IF(IK.EQ.1) GO TO 50
      WRITE(6,36)TOT ,TEE,PCMEK,VEREL,CMAA,PLABE
   36 FORMAT(6(F9.4,2X))
      WRITE(6,30)
   30 FORMAT(1HO' CM ANGLE  L.ANGLE 1 ENER 1    L.ANGLE 2 ENER 2      DET
     1.      TRANSF. ')
      IK=1
   50 CONTINUE
      SCANA=SCAN+SSMAL
      CALL SCATT(PTX,PBK,ENK,ENT,PTZ,SCANA,FINA,FITA,FANA,FAN2A,
     1PCMEA,CMANA)
      DFADS=(COS(FANA)-COS(FANG))/(COS(CMAN+SCAN)-COS(CMAN+SCANA))
      DF2DS=(COS(FAN2A)-COS(FANG2))/(COS(CMAN+SCAN)-COS(CMAN+SCANA))
      PTGTC=PTGT+PSMAL
      CALL NUCSYS(PTGTC,ENKC,PBKC,BETC,ENTGC,ENRCC)
      CALL DISSOC(PTGTC,THTG,BETC,ENTGC,ENRCC,PTZC,PTXC,ENTC)
      CALL SCATT(PTXC,PBKC,ENKC,ENTC,PTZC,SCAN,FINC,FITC,FANC,
     1FAN2C,PCMEC,CMANC)
      DFEDP=(FINC+FITC-FINEN-FITEN)/PSMAL
      DFADP=(COS(FANC)-COS(FANG))/PSMAL
      DF2DP=(COS(FAN2C)-COS(FANG2))/PSMAL
      THTB=THTG+TSMAL
      IF(PTGT.NE.0.0)GO TO 150
      PTGTB=PSMAL
      CALL NUCSYS(PTGTB,ENKB,PBKB,BETB,ENTGB,ENRCB)
      CALL DISSOC(PTGTB,THTB,BETB,ENTGB,ENRCB,PTZB,PTXB,ENTB)
      CALL SCATT(PTXB,PBKB,ENKB,ENTB,PTZB,SCAN,FINB,FITB,FANB,FAN2B,
     1PCMEB,CMANB)
      DFEDT=(FINB+FITB-FINC-FITC)/(COS(THTG)-COS(THTB))
      DFADT=(COS(FANB)-COS(FANC))/(COS(THTG)-COS(THTB))
      DF2DT=(COS(FAN2B)-COS(FAN2C))/(COS(THTG)-COS(THTB))
      GO TO 151
  150 CALL DISSOC(PTGT,THTB,BETFR,ENTGT,ENRC,PTZB,PTXB,ENTB)
      CALL SCATT(PTXB,PBK,ENK,ENTB,PTZB,SCAN,FINB,FITB,FANB,FAN2B,
     1PCMEB,CMANB)
      DFEDT=(FINB+FITB-FINEN-FITEN)/(COS(THTG)-COS(THTB))
      DFADT=(COS(FANB)-COS(FANG))/(COS(THTG)-COS(THTB))
      DF2DT=(COS(FAN2B)-COS(FANG2))/(COS(THTG)-COS(THTB))
  151 TRJAC=DFEDP*DFADT*DF2DS+DFEDT*DFADS*DF2DP-DFEDP*DFADS*DF2DT-
     1DFEDT*DFADP*DF2DS
      FASE=PTGT**2/TRJAC
      IF(PTGT.EQ.0.0) FASE=PTGTB**2/TRJAC
      WRITE(6,31)SCNA,FNGA,FINEK,FAA2A,FITEK,TRJAC,FASE
   31 FORMAT(2(F7.2,3X),F9.4,1X,F7.2,1X,F9.4,2(1X,E11.4))
      IF(SCAN.LT.SCANM)GO TO 20
      SCAN=SCANO
      IF(PTGT.EQ.0.0)GO TO 15
```

```
      IF(THTG.LT.THTGM)GO TO 19
      THTG=THTGO
      IF(PTGT.LT.ENKTM)GO TO 15
  100 CONTINUE
      GO TO 1
      END
```

```
      SUBROUTINE MMULT (AA,AB,AC)
C PROGRAM MULTIPLIES THE 4-DIM. MATRICES AA(IA,JA) AND AB(IB,JB)
      IMPLICIT COMPLEX*8(A)
      DIMENSION AA(4,4),AB(4,4),AC(4,4)
      DO 10 IA=1,4
      DO 10 JB=1,4
      AC(IA,JB)=AA(IA,1)*AB(1,JB)+AA(IA,2)*AB(2,JB)+AA(IA,3)*AB(3,JB)+
     1AA(IA,4)*AB(4,JB)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE NUCSYS(PTGT,ENK,PBK,BETFR,ENTGT,ENRC)
      IMPLICIT REAL*4(M)
      COMMON MASIN,MASTG,MASCL,MASR,ENINM,ENINT,ENMOV,PMOV,BETI,BETM,
     1PT,THT,PNX,PNZ,THN,PN,ENN,ET,PX,EX,FINMX,FINMZ,PMOMI
     2,VREL
      ENTGT=SQRT(PTGT**2+MASCL**2)
      ENRC=SQRT(PTGT**2+MASR**2)
      VAR1=MASIN**2+MASTG**2+2.0*MASTG*ENINM
      ADA=PTGT**2+0.5*(MASR**2-MASTG**2-2.0*MASIN**2+MASCL**2)
     1-ENINM*MASTG+ENTGT*ENRC
      ENK=(PMOMI*SQRT(ADA**2-VAR1*MASIN**2)-ADA*ENINT)/VAR1
      PBK=SQRT(ENK**2-MASIN**2)
      ENMOV=(ENINT**2-PMOMI**2- 2.0*ENTGT**2-MASR**2-2.0*ENTGT*ENRC+
     1MASCL **2-MASIN**2)/(2.0*(ENTGT+ENRC))
      PMOV=SQRT(ENMOV**2-MASIN**2)
      BETI=PBK/ENK
      BETM=PMOV/ENMOV
      BETFR=(BETI-BETM)/(1.0-BETI*BETM)
      RETURN
      END
```

```
      SUBROUTINE DISSOC(PTGT,THTG,BETFR,ENTGT,ENRC,PTZ,PTX,ENT)
      IMPLICIT REAL*4(M)
      COMMON MASIN,MASTG,MASCL,MASR,ENINM,ENINT,ENMOV,PMOV,BETI,BETM,
     1PT,THT,PNX,PNZ,THN,PN,ENN,ET,PX,EX,FINMX,FINMZ,PMOMI
     2,VREL
      GAM=SQRT(1.0/(1.0-BETFR**2))
      PX=PMOV+BETFR*GAM*(GAM*BETFR*PMOV/(1.0+GAM)+ENMOV)
      EX=SQRT(PX**2+MASIN**2)
      PTZ=PTGT*COS(THTG)+BETFR*GAM*(GAM*BETFR*PTGT*COS(THTG)/(1.0+GAM)+
     1ENTGT)
      PTX=PTGT*SIN(THTG)
      PT=SQRT(PTZ**2+PTX**2)
      IF(PTX.EQ.0.0.AND.PTZ.EQ.0.0) PTZ=0.00001
      THT=ATAN2(PTX,PTZ)
      ENT=SQRT(PT**2+MASCL**2)
      PNX=PTGT*SIN(3.141593+THTG)
      PNZ=PTGT*COS(3.141593+THTG)+BETFR*GAM*(GAM*BETFR*PTGT*
     1COS(3.141593+THTG)/(1.0+GAM)+ENRC)
      IF(PNX.EQ.0.0.AND.PNZ.EQ.0.0)PNZ=0.00001
      THN=ATAN2(PNX,PNZ)
      PN=SQRT(PNZ**2+PNX**2)
      ENN=SQRT(PN**2+MASR**2)
      RETURN
      END
```

```
      SUBROUTINE SCATT(PTX,PBK,ENK,ENT,PTZ,SCAN,FINEN,FITEN,FANG,
     1FANG2,PCMEN,CMAN)
      IMPLICIT COMPLEX*8(A) ,REAL*4(M)
      COMMON MASIN,MASTG,MASCL,MASR,ENINM,ENINT,ENMOV,PMOV,BETI,BETM,
     1PT,THT,PNX,PNZ,THN,PN,ENN,ET,PX,EX,FINMX,FINMZ,PMOMI
     2,VREL
      DIMENSION APK(4,4),APT(4,4),AA(4,4),APKCM(4,4),APTCM(4,4),
     1AROT(4,4),AFINM(4,4),AFINT(4,4), APTSC(4,4),APKSC(4,4)
      BETAX=-PTX/(ENK+ENT)
      BETAZ=-(PBK+PTZ)/(ENK+ENT)
      BETSQ=BETAZ**2+BETAX**2
      APK(1,1)=(0.0,0.0)
      APK(2,1)=(0.0,0.0)
      APK(3,1)= CMPLX(0.0,PBK)
      APK(4,1)=CMPLX(ENK,0.0)
      APT(1,1)=CMPLX(0.0,PTX)
      APT(2,1)=(0.0,0.0)
      APT(3,1)=CMPLX(0.0,PTZ)
      APT(4,1)=CMPLX(ENT,0.0)
      DO 5 IP=1,4
      DO 5 JP=2,4
      APK(IP,JP)=(0.0,0.0)
    5 APT(IP,JP)=(0.0,0.0)
      GAMM=1.0/SQRT(1.0-BETSQ)
      HAMM=GAMM-1.0
      INDEX=0
   10 AA(1,1)=CMPLX(1.0+HAMM*BETAX**2/BETSQ,0.0)
      AA(1,2)=(0.0,0.0)
      AA(1,3)=CMPLX(BETAX*BETAZ*HAMM/BETSQ,0.0)
      AA(1,4)=CMPLX(0.0,BETAX*GAMM)
      AA(2,1)=(0.0,0.0)
      AA(2,2)=(1.0,0.0)
      AA(2,3)=(0.0,0.0)
      AA(2,4)=(0.0,0.0)
      AA(3,1)=AA(1,3)
      AA(3,2)=(0.0,0.0)
      AA(3,3)=CMPLX(1.0+HAMM*BETAZ**2/BETSQ,0.0)
      AA(3,4)=CMPLX(0.0,BETAZ*GAMM)
      AA(4,1)=CMPLX(0.0,-BETAX*GAMM)
      AA(4,2)=(0.0,0.0)
      AA(4,3)=CMPLX(0.0,-BETAZ*GAMM)
      AA(4,4)=CMPLX(GAMM,0.0)
      IF(INDEX.EQ.1) GO TO 20
      CALL MMULT(AA,APK,APKCM)
      CALL MMULT(AA,APT,APTCM)
      PCMEN=REAL(APKCM(4,1))+REAL(APTCM(4,1))
      PKCMX=AIMAG(APKCM(1,1))
      PKCMZ=AIMAG(APKCM(3,1))
      IF(PKCMX.EQ.0.0.AND.PKCMZ.EQ.0.0)PKCMZ=.00001
```

```
      CMAN=ATAN2(PKCMX,PKCMZ)
      VEE1=(SQRT(PKCMX**2+PKCMZ**2))/REAL(APKCM(4,1))
      VEE2=(SQRT(PKCMX**2+PKCMZ**2))/REAL(APTCM(4,1))
      VREL=(VEE1+VEE2)/(1.0+VEE1*VEE2)
      AROT(1,1)=CMPLX(COS(SCAN),0.0)
      AROT(1,2)=(0.0,0.0)
      AROT(1,3)=CMPLX(SIN(SCAN),0.0)
      AROT(1,4)=(0.0,0.0)
      AROT(2,1)=(0.0,0.0)
      AROT(2,2)=(1.0,0.0)
      AROT(2,3)=(0.0,0.0)
      AROT(2,4)=(0.0,0.0)
      AROT(3,1)=CMPLX(-SIN(SCAN),0.0)
      AROT(3,2)=(0.0,0.0)
      AROT(3,3)=CMPLX(COS(SCAN),0.0)
      AROT(3,4)=(0.0,0.0)
      AROT(4,1)=(0.0,0.0)
      AROT(4,2)=(0.0,0.0)
      AROT(4,3)=(0.0,0.0)
      AROT(4,4)=(1.0,0.0)
      CALL MMULT(AROT,APKCM,APKSC)
      CALL MMULT(AROT,APTCM,APTSC)
      BETAX=-BETAX
      BETAZ=-BETAZ
      INDEX=1
      GO TO 10
   20 CONTINUE
      CALL MMULT(AA,APKSC,AFINM)
      CALL MMULT(AA,APTSC,AFINT)
      FINMX=AIMAG(AFINM(1,1))
      FINMZ=AIMAG(AFINM(3,1))
      FINEN=REAL(AFINM(4,1))
      FINTX=AIMAG(AFINT(1,1))
      FINTZ=AIMAG(AFINT(3,1))
      FITEN=REAL(AFINT(4,1))
      FANG=ATAN2(FINMX,FINMZ)
      FANG2=ATAN2(FINTX,FINTZ)
      RETURN
      END
```