JAERI-Data/Code
2001-007

HIGH ENERGY PARTICLE TRANSPORT CODE NMTC/JAM

March 2001

Koji NIITA*, Shin-ichiro MEIGO, Hiroshi TAKADA and Yujiro IKEDA

日本原子力研究所
Japan Atomic Energy Research Institute

本レポートは、日本原子力研究所が不定期に公刊している研究報告書です。
入手の問合わせは、日本原子力研究所研究情報部研究情報課（〒319-1195　茨城県那珂郡東海村）あて、お申し越し下さい。なお、このほかに財団法人原子力弘済会資料センター（〒319-1195　茨城県那珂郡東海村日本原子力研究所内）で複写による実費頒布を行っております。

This report is issued irregularly.
Inquiries about availability of the reports should be addressed to Research Information Division, Department of Intellectual Resources, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken 〒319-1195, Japan.

編集兼発行　日本原子力研究所

JAERI－Data/Code 2001－007

High Energy Particle Transport Code NMTC/JAM

Koji NIITA*, Shin-ichiro MEIGO, Hiroshi TAKADA and Yujiro IKEDA

Center for Neutron Science
Tokai Research Establishment
Japan Atomic Energy Research Institute
Tokai-mura, Naka-gun, Ibaraki-ken

(Received January 30, 2001)

We have developed a high energy particle transport code NMTC/JAM, which is an upgraded version of NMTC/JAERI97. The applicable energy range of NMTC/JAM is extended in principle up to 200 GeV for nucleons and mesons by introducing the high energy nuclear reaction code JAM for the intra-nuclear cascade part. For the evaporation and fission process, we have also implemented a new model, GEM, by which the light nucleus production from the excited residual nucleus can be described. According to the extension of the applicable energy, we have upgraded the nucleon-nucleus non-elastic, elastic and differential elastic cross section data by employing new systematics. In addition, the particle transport in a magnetic field has been implemented for the beam transport calculations. In this upgrade, some new tally functions are added and the format of input and output of data has been improved very much in a user friendly manner. Due to the implementation of these new calculation functions and utilities, consequently, NMTC/JAM enables us to carry out reliable neutronics study of a large scale target system with complex geometry more accurately and easily than before. This report serves as a user manual of the code.

Keywords: NMTC/JAM, Intranuclear Cascade, JAM, Nucleon-Nucleus Cross Sections, GEM (Generalized Evaporation Model), Magnetic Field, Beam Transport

* Research Organization for Information Science & Technology

JAERI－Data/Code 2001－007

高エネルギー粒子輸送コード NMTC/JAM

日本原子力研究所東海研究所中性子科学研究センター
仁井田 浩二* ・ 明午 伸一郎 ・ 高田 弘 ・ 池田 裕二郎

(2001 年 1 月 30 日受理)

NMTC/JAERI97 コードの改良版として、高エネルギー粒子輸送コード NMTC/JAM を開発した。 NMTC/JAM は、核内カスケードモデルとして、高エネルギー核反応コード JAM を導入することにより、その適用エネルギー範囲を原理的には、核子、中間子に対して 200GeV まで引き上げた。また、蒸発、核分裂過程に対して GEM モデルを導入することにより、励起した残留核からの軽核生成の記述が可能になった。適用エネルギーの拡張にともない、核子-原子核の非弾性散乱断面積、弾性散乱断面積、弾性散乱断面積の角分布のデータを新しい系統式を用いて更新した。更に、ビーム輸送計算に必要な磁場中の粒子輸送を計算できるようにした。また、幾つかのタリー機能が追加され、データの入出力については、ユーザーの利便性を高めるために大きく改良された。このような新しい物理モデルとユーティリティーが導入されたことにより、NMTC/JAM は複雑な体系の大きなターゲットシステムのニュートロニクス計算に対して、以前よりも信頼性の高い結果を与えることができるようになった。この報告は、NMTC/JAM コードのユーザーマニュアルである。

東海研究所：〒 319-1195 茨城県那珂郡東海村白方白根 2-4
* (財) 高度情報科学技術研究機構

Contents

目　次

1. Introduction

The nucleon meson transport code NMTC/JAERI [1] was developed in 1982 from its original version of NMTC [2] by Nakahara and Tsutsui for studying nuclear spallation reactions and particle transport in a medium. The code can simulate both the primary spallation reactions and the secondary particle transport in the intermediate energy region from 20 MeV to 3.5 GeV by the use of the Monte Carlo technique. In this code, nuclear reactions had been simulated by the intranuclear cascade [3] and evaporation [4] model including high energy fission process [5].

In 1997, we upgraded the physical models and nuclear data employed in NMTC/JAERI code to improve its own simulation accuracy, and we named this code system NMTC/JAERI97 [6]. In this upgrading works, we introduced a preequilibrium model based on the exciton model, precise level density parameters dependent on an excitation energy of a residual nucleus, and an alternative intranuclear cascade model in which nuclear medium effects were taken into account in terms of reflection and refraction with in-medium nucleon-nucleon cross sections. We also upgraded nucleon-nucleus cross sections by use of the systematics derived by Pearlstein [7] from the geometrical cross section of a target nucleus. A new geometry package based on the combinatorial geometry (CG) method, MARS-CG [8, 9], was implemented for describing complex configurations such as a target-moderator-reflector system for intense spallation neutron source. Several tally functions and the importance sampling technique were also developed for an easy handling of the final results and for variance reduction of the Monte Carlo calculation of large scale systems, respectively.

Recently, Japan Atomic Energy Research Institute (JAERI) and High Energy Accelerator Research Organization (KEK) have jointly proposed high intensity proton accelerator project for comprehensive studies on materials and life science, elemental particle and nuclear physics and basic technology development of accelerator-driven transmutation [10] systems. In this project, a neutronics design study has been conducted at JAERI to construct a research complex utilizing a high-intensity pulsed spallation neutron source driven by 3 GeV proton beams with intensity up to 1 MW.

In the neutronics design study, the NMTC/JAERI97 code has been used in combination with MCNP4A [11] as a standard code system at JAERI. For the optimization study of the spallation neutron source facility, validation of NMTC/JAERI97 has been carried out to confirm its reliability. Experimental studies have been performed at the AGS facility of Brookhaven National Laboratory under the ASTE (AGS Spallation Target Experiment) collaboration framework [12] using a mercury target bombarded with protons at incident energies ranging from 1.6 to 24 GeV.

In the analytical study of this experiment, it is requested to describe the physical process to produce neutrons up to 24 GeV as precise as possible. Since NMTC/JAERI97 cannot treat the nuclear reactions above 3.5 GeV, it is required to extend the applicable energy range to higher region. It is also required for neutronics design study of elemental particle and nuclear

physics facilities in the high intensity proton accelerator project where the maximum 50 GeV proton beam is planned. To meet the requirements, we have introduced the high energy nuclear reaction code JAM [13] into the intra-nuclear cascade part of NMTC/JAERI97 and we have named this new code system NMTC/JAM. In the NMTC/JAM code, we have also upgraded the nucleon-nucleus non-elastic, elastic and differential elastic cross section data by employing a new systematics. The GEM model [14] has been also implemented for the evaporation and fission model, which can describe the light-ion production like Be isotopes from the excited residual nucleus. In addition, particle transport in a magnetic field has been implemented for the beam transport calculations.

In this report, we describe the overview of the NMTC/JAM code in section 2., the models implemented in the code in section 3. The geometry configuration is explained in section 4, the importance sampling technique in section 5. Sections 6 is devoted to the tally functions. New function of the particle transport in a magnetic field and the nuclear reaction calculation are introduced in sections 7 and 8. In section 9, we give the user guide for installation of code, preparation of the input data, execution of the calculation. Section 10 shows several examples of execution of NMTC/JAM. We summarize this report in section 11.

2. Overview of Code

NMTC/JAM simulates the transport in a medium of the particles of nucleons, mesons, barions and leptons as well as their anti-particles, which are listed in Table 1, using the Monte Carlo technique. Any target nucleus can be used for the material in a medium. However, the light ions like deuteron, triton, ^{3}He and α and heavy ions are not considered as transport particles.

Table 1. List of the transport particles.

ityp	symbol	kf-code	particle name
1	proton	2212	proton
2	neutron	2112	neutron
3	pion+	211	π^+
4	pion0	111	π^0
5	pion-	-211	π^-
6	mu+	-13	μ^+
7	mu-	13	μ^-
8	kaon+	321	K^+
9	kaon0	311	K^0
10	kaon-	-321	K^-
14	-	22	γ
11	-	+ - 12	ν_e $\overline{\nu_e}$
11	-	+ - 14	ν_μ $\overline{\nu_\mu}$
11	-	-2212	$\overline{p}$
11	-	-2112	$\overline{n}$
11	-	-311	$\overline{K^0}$
11	-	+ - 221	η $\overline{\eta}$
11	-	331	η'
11	-	+ - 3122	Λ^0 $\overline{\Lambda^0}$
11	-	+ - 3222	Σ^+ $\overline{\Sigma^+}$
11	-	+ - 3212	Σ^0 $\overline{\Sigma^0}$
11	-	+ - 3112	Σ^- $\overline{\Sigma^-}$
11	-	+ - 3322	Ξ^0 $\overline{\Xi^0}$
11	-	+ - 3312	Ξ^- $\overline{\Xi^-}$
11	-	+ - 3334	Ω^- $\overline{\Omega^-}$

The random number generator based on the congruential scheme of Lehmer [15], which is the same as for the MCNP4A [11] code, is employed for the Monte Carlo calculation. By the input parameters, one can change the initial seed of random number and skip a certain number of random number sequences. So the user is able to repeat the same calculation, or reproduce a

part of the calculation in a easy way.

For the transport calculation, the maximum energy is limited to about 200 GeV for all transport particles, while the minimum energy is restricted to 20 MeV for neutrons, 0.1 MeV for the other particles, respectively. The photo-nuclear reactions are, however, not taken into account. These limitations are due to the limit of the nuclear reaction models implemented in the code. As for the cutoff neutrons below the limit, and the cutoff photons below the threshold energy defined by the user, the phase space information of these cutoff particles at the cutoff point can be stored in the file for sequential calculations. The transport of the low energy neutrons and photons can be calculated by the neutron-photon transport code such as MCNP4A using a cross section library processed from evaluated nuclear data. Figure 1 shows the flow of neutronics calculation with the NMTC/JAM and the MCNP4A code. A user-supplied source routine is required to read the information of the cutoff neutrons and photon for the MCNP4A calculation.

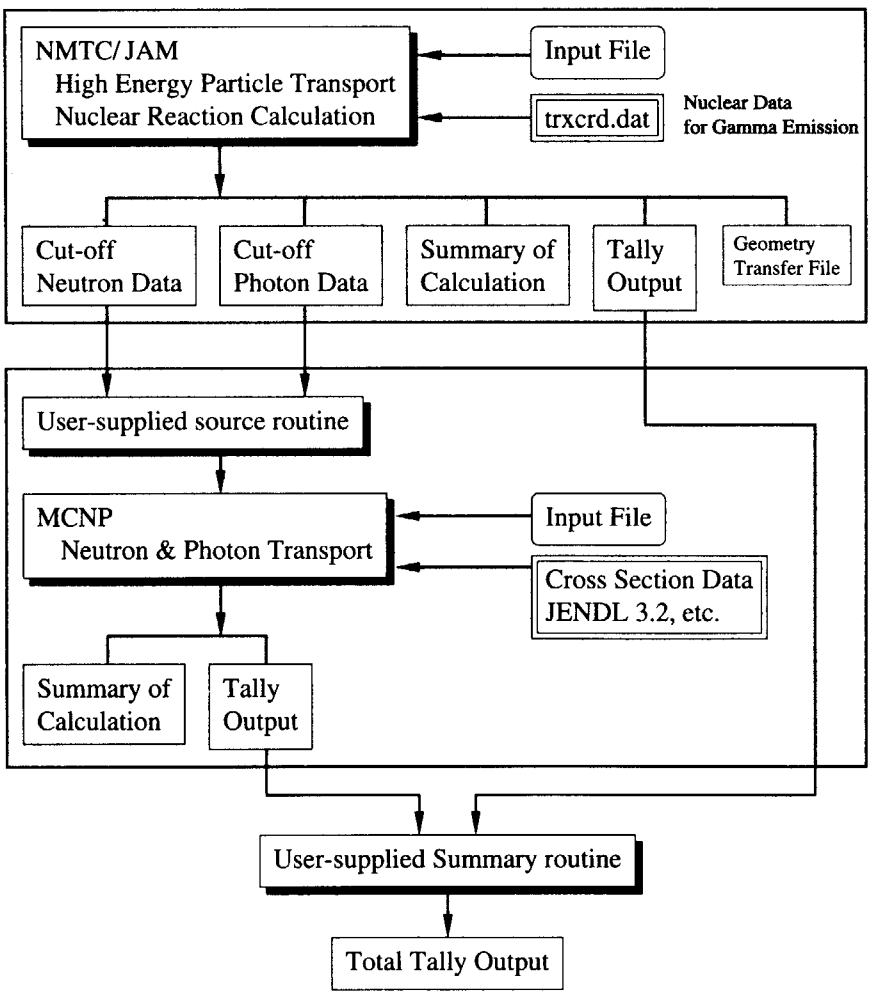

Figure 1. Flow of neutronics calculation with the NMTC/JAM and MCNP4A code.

2.1. Main Routine

The main routine of the NMTC/JAM code is rather simple. We show the program flow of the main routine in Figure 2, where the name enclosed by a box is the name of subroutine called from main routine. In the main routine, we call the subroutines which read the input data file and initialize all components of the code. After the initialization, it calls two subroutines, **ovly12** and **ovly13**, which are the main part of the simulation of the particle transport in a medium, and the nuclear reactions, which are selected by the input parameter.

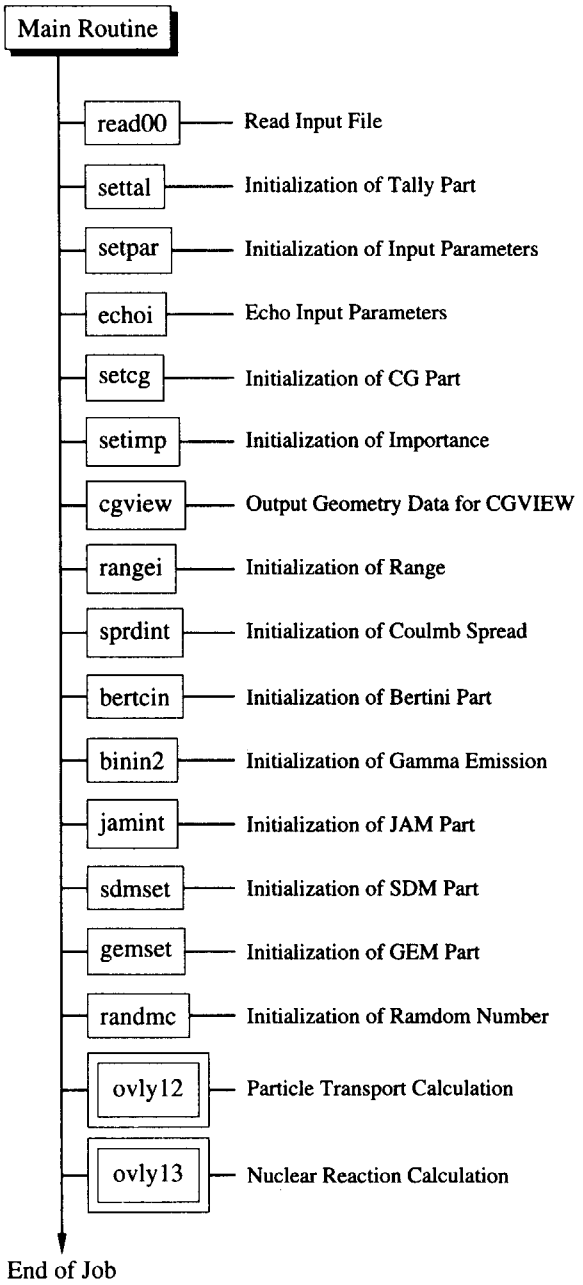

Figure 2. Flow of main routine.

2.2. The Control Routine of Particle Transport: ovly12

The subroutine **ovly12** is a main control routine of the particle transport in a medium by the Monte Carlo method. In this subroutine, we generate a source particle by the subroutine **sors** and transport the particle until all particles, which are produced sequentially in the transportation by the initial particle, are dead by the energy cutoff or by escaping from the system. Figure 3 shows the algorithm flow of the subroutine **ovly12**.

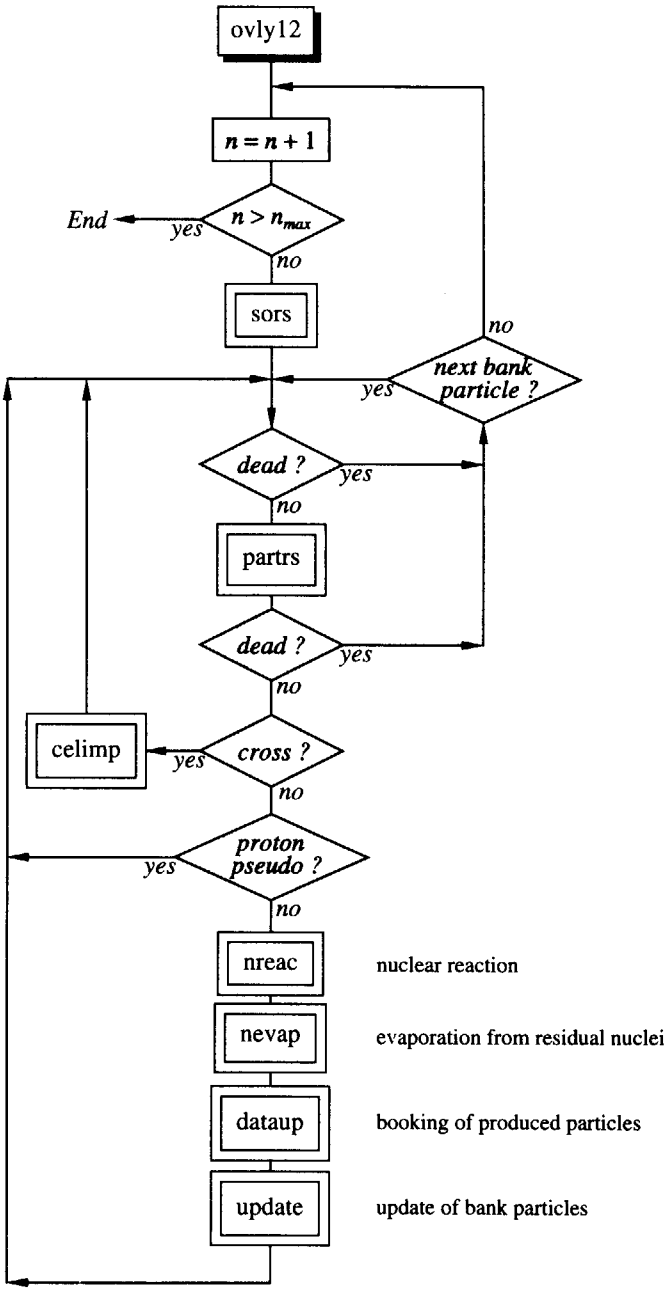

Figure 3. Flow of the subroutine **ovly12**.

In this figure, n_{max} denotes the maximum number of source particles which should be calculated. The source particle generated by **sors** goes into the subroutine **partrs**, which transports the particle up to the next collision point or the next boundary point of the region, losing the energy by the ionization in the material for charged particle, under the magnetic field or Coulomb spreading field if exists.

After **partrs**, we check the energy of the particle. If the energy is lower than the cutoff energy, this particle is dead, and we look for next particle in the bank, where the produced particles are stored and waiting its transportation. If the energy is above the cuttoff energy, we check the position of the particle. If the particle crosses the boundary of the region, the particle goes back to **partrs** taking into account the tally functions and the importance function if the user requires.

If the particle reaches the collision point, we consider the nuclear reactions with the target nucleus of the material. For protons, we use the pseudo collision procedure, since the energy decreases from the starting point to the collision point due to the ionization loss while the nucleon-nucleus cross section depends on the energy.

The nuclear reactions are treated by the subroutines **nreac** and **nevap**. The former describes the dynamical part of the reactions, while the latter the evaporation, fission and gamma emission of the residual nuclei. After the nuclear reactions, we book the produced particles into the bank and update the bank particles, by the subroutines **dataup** and **update**. Finally we consider the tally functions, and go back to the next particle transport.

2.3. Subroutine of Particle Transport: partrs

In subroutine **partrs**, we first estimate a mean free path from the life time of the particles and the cross sections of the particle and the nuclei of the material. From the mean free path, we determine the flight path length, fpl, by a random sampling of exponential probability distribution. These procedure is executed in the subroutine **getflt** in Figure 4.

The particle transport in the medium of the flight path length fpl, is calculated by the subroutines, **sprd**, **magfld** and **parfre**. The subroutine **sprd** describes the proton beam transfer with Coulomb spread, **magfld** the charged particle transfer in a magnetic field, and **parfre** calculates the particle transfer without any field in a medium, respectively. These three subroutines have a similar structure, which is shown in Figure 5.

For the cases with the external field like magnetic field, it is difficult to determine the next cross point of the region, since the trajectory is not a straight line. Therefore, we define the minimum step of the transfer distance d_{min} and search the coordinate of the cross point as precisely as possible. We use 0.1 cm for d_{min} as a default value in the code, which is able to be changed by the input parameter.

In Figure 5, d denotes the temporary flight path length to be transferred, while d_r is the remaining distance. By the subroutine **rainge**, we estimate the range r for the charged particle,

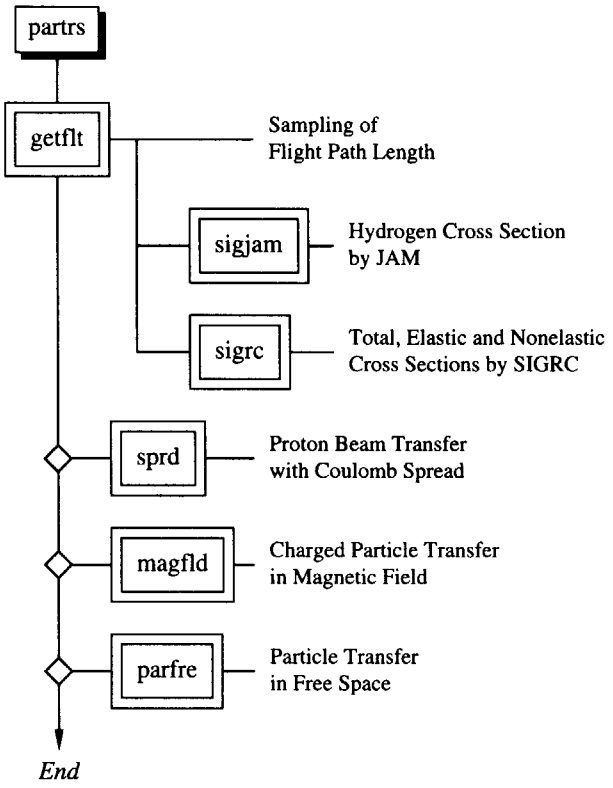

Figure 4. Flow of the subroutine **partrs**.

which is a flight distance until the particle stops in the medium.

According to the range r, we modify the d, and check whether the d is larger than d_{min} or not. For the charged particle, the energy is corrected by the subroutine **ecol**, and we transfer the particle by the distance d in the subroutines **free**, **sprdtrs**, **magtrs**, respectively. By these subroutines, we determine the final point and final momentum of the particle after the propagation of the distance d.

Then we check the final position by the subroutine **gomprp**. If the particle crosses the boundary of the region and d already reaches d_{min}, the process is finished after the energy of the charged particle is corrected by **ecol**. For the cross case, but d is still larger than d_{min}, we reduce the value of d by 10 % and go back to the starting point. If the particle does not cross the boundary and the remaining flight length is zero, the process is finished. But for the case that the remaining flight length still exists, we go back to the next step.

Above procedure is not so simple, i.e. we transfer the particle by searching the crossing point with changing the value d. However, if we use the fixed value of d, the calculation time increases drastically with small value of d and for the large system. On the other hand, the resolution of the calculation becomes too broad with large value of d compared with the typical scale of the system. Therefore, above procedure satisfies enough accuracy as well as quick calculation time.

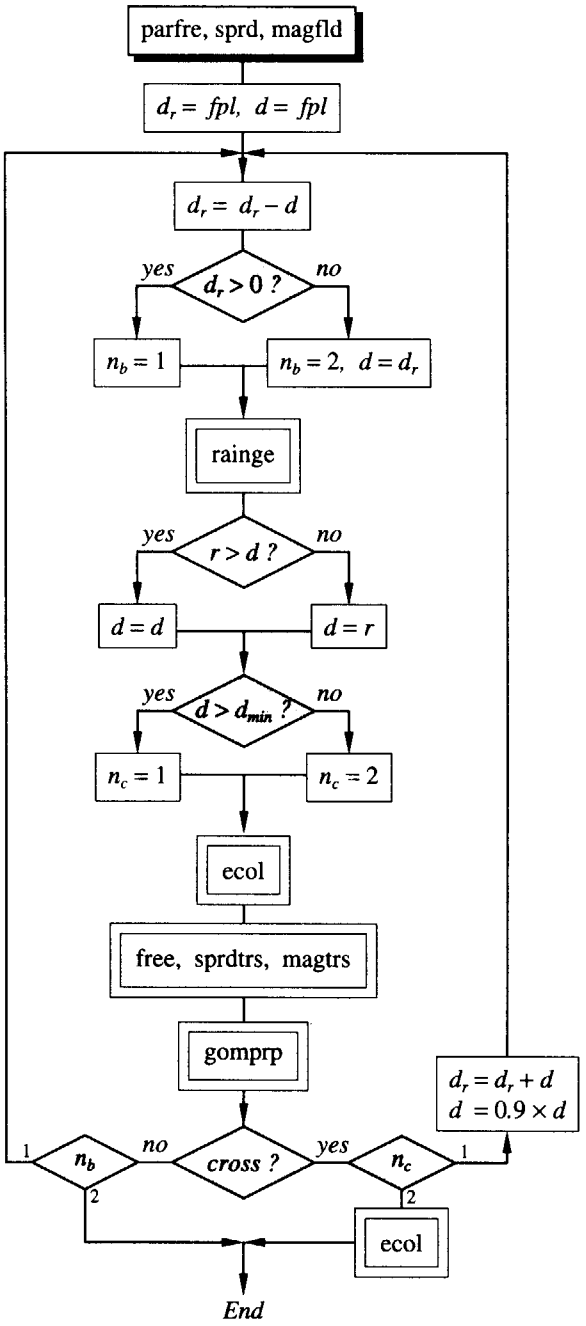

Figure 5. Flow of the subroutine **parfre, sprd, magfld**.

2.4. Subroutine of Nuclear Reaction: nreac

Nuclear reaction part consists of four processes, collision with Hydrogen, particle decay, elastic collision, and non-elastic collision, which are calculated by the subroutines **jamin**, **dklos**, **nelst**, and **ncasc** in the subroutine **nreac** shown in Figure 6.

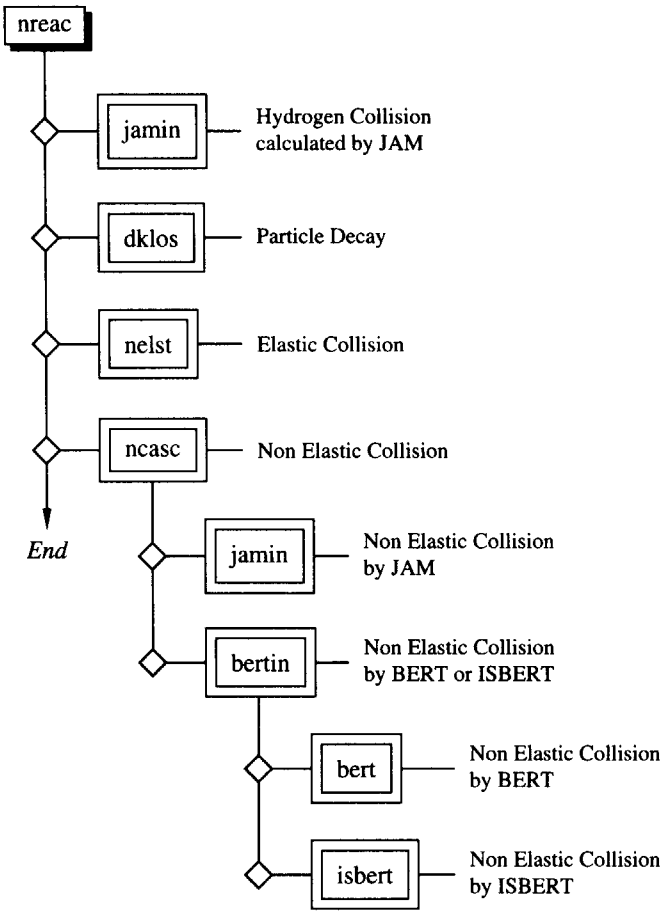

Figure 6. Flow of the subroutine **nreac**.

For the non-elastic collision, we have implemented three models of intranuclear cascade calculation, JAM model, Bertini model and Isobar model. For a default input, the non-elastic collision is calculated by JAM model above 3.5 GeV, by Bertini model below 3.5 GeV. By the input parameters, the user chooses the models and the threshold energies of switching models. The details of these models are discussed in the next section.

2.5. Subroutine of Evaporation: nevap

After the nuclear reaction part, we consider the evaporation, fission, and gamma emission from the excited residual nuclei. We have three models of the fission and evaporation processes, SDM model, GEM model and ERUP package. The last package is a model implemented in NMTC/JAERI97 code. These are controlled by the subroutines, **sdmexec**, **gemexec** and

erupin in Figure 7. After these fission and evaporation process, if the excitation energy still remains, we consider the gamma emission from the residual nuclei by subroutine **dexgam**. The details of these models are discussed in the next section.

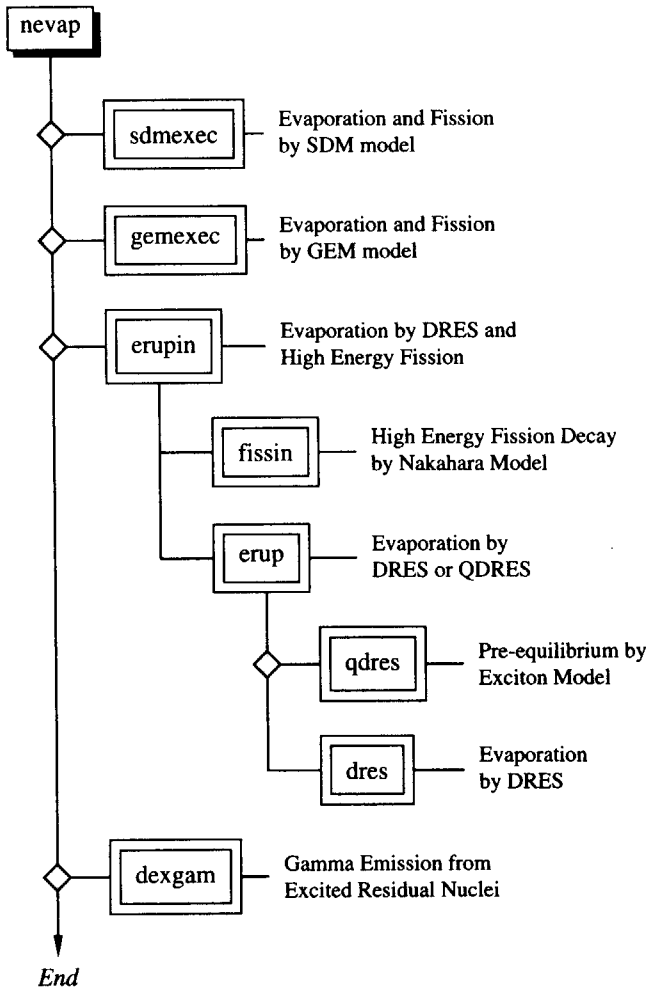

Figure 7. Flow of the subroutine **nevap**.

3. Models Implemented in the Code

3.1. JAM model

3.1.1 Main Features of JAM

JAM (Jet AA Microscopic Transport Model) [13] is a hadronic cascade model, which explicitly treats all established hadronic states including resonances with explicit spin and isospin as well as their anti-particles. We have parametrized all hadron-hadron cross sections based on the resonance model and string model by fitting the available experimental data. Below center of mass energy $\sqrt{s} < 4$ GeV, the inelastic hadron-hadron collisions are described by the resonance formations and their decays, and at higher energies, string formation and their fragmentation into hadrons are assumed.

We have parametrized the resonance formation cross sections in terms of the extended Breit-Wigner form and used the established data [16] for its decay channels and probabilities. At an energy range above $\sqrt{s} > 4 \sim 5$ GeV, the (isolated) resonance picture breaks down because width of the resonance becomes wider and the discrete levels get closer. The hadronic interactions at the energy range $4 \sim 5 < \sqrt{s} < 10 \sim 100$ GeV where it is characterized by the small transverse momentum transfer is called "soft process", and string phenomenological models are known to describe the data for such soft interaction well. The hadron-hadron collision leads to a string like excitation longitudinally. In actual description of the string formation, we follow the prescription adopted in the HIJING model [17]. The strings are assumed to hadronize via quark-antiquark or diquark-antidiquark creation. As for the fragmentation of the strings, we adopted Lund fragmentation model PYTHIA6.1 [18].

In Figure 8, we show the fitted total cross section with experimental data [16] and inelastic components of pp collision as a function of c.m. energy. Inelastic cross sections are assumed to be filled up with the resonance formations (gray region) up to $\sqrt{s} = 3 \sim 4$ GeV. At higher energies, the difference between experimental inelastic cross section and resonance formation cross sections are assigned to the string formation. The following resonance excitation channels are implemented for the nucleon-nucleon scattering in JAM:

(1) $NN \to N\Delta(1232)$, (2) $NN \to NN^*$, (3) $NN \to \Delta(1232)\Delta(1232)$,
(4) $NN \to N\Delta^*$, (5) $NN \to N^*\Delta(1232)$, (6) $NN \to \Delta(1232)\Delta^*$,
(7) $NN \to N^*N^*$, (8) $NN \to N^*\Delta^*$, (9) $NN \to \Delta^*\Delta^*$.

Here N^* and Δ^* represent higher baryonic states below 2 GeV/c^2. In the same figure, we also plot the contribution from the above channels (1) (dashed line), (2) (dot-dot-dashed line), (4) (long dashed line) and the sum of the other channels (dot-dashed line) to the resonance formation cross section.

For nuclear reactions in JAM, we use the full cascade method described in the following.

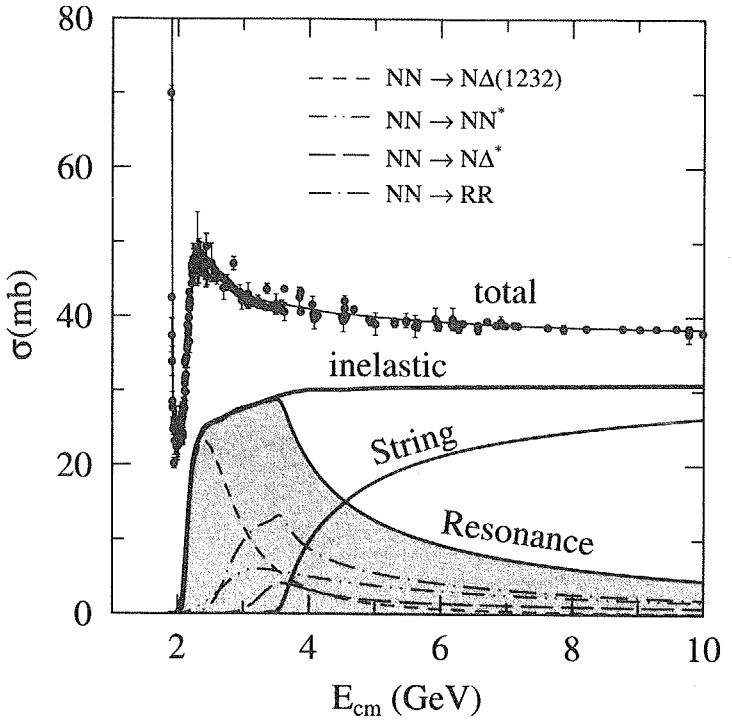

Figure 8. The fitted total cross section and inelastic components of *pp* collision as a function of c.m. energy.

Each hadron has its position and momentum and moves along a straight line until it meets next hadron-hadron collision, decay or absorption. The initial position of each nucleon is sampled by the parameterized distribution of nuclear density. Fermi momentum of nucleons is assigned according to the local Fermi momentum as a function of the density. We do not take into account the mean field effects except for the initial nucleons. The initial nucleons in target nucleus stay on the initial positions until the collision with the other hadrons. The interaction probabilities of hadron-hadron are determined by the method of so-called "closest distance approach"; if the minimum relative distance for any pair of particles becomes less than interaction range specified by $\sqrt{\sigma(\sqrt{s})/\pi}$, where $\sigma(\sqrt{s})$ is the total cross section for the pair at the c.m. energy $\sqrt{s}$, then particles are assumed to collide. This cascade method has been widely used to simulate high energy nucleus-nucleus collisions. However, geometrical interpretation of the cross section violates causality and the time ordering of the collisions in general differs from one reference-frame to another. These problems have been studied by several authors [19, 20]. We have adopted the similar procedure as that in Refs. [19, 21] for the collision criterion to mimic the reference-frame dependence. Pauli-blocking for the final nucleons in two-body collisions are also considered. For the comparison with the alternative methods of the cascade, we have compared the results of JAM with that of Glauber type calculations in Ref. [13].

3.1.2 Elementary Cross Sections of Hadron-Hadron

The detail of the parametrization of hadron-hadron cross sections in JAM is described in Ref. [13]. Here, we demonstrate the typical examples of the elementary hadron-hadron cross sections obtained by JAM and compare results with the experimental data.

In Figure 9 we show the calculated rapidity y distributions and the transverse momentum distributions of protons, positive and negative pions for proton-proton collisions at 12GeV/c and also the data from Ref. [22]. The proton stopping behavior and the pion yields are well described by the JAM. Within JAM model, fast protons come from resonance decays and mid-rapidity protons from string fragmentation.

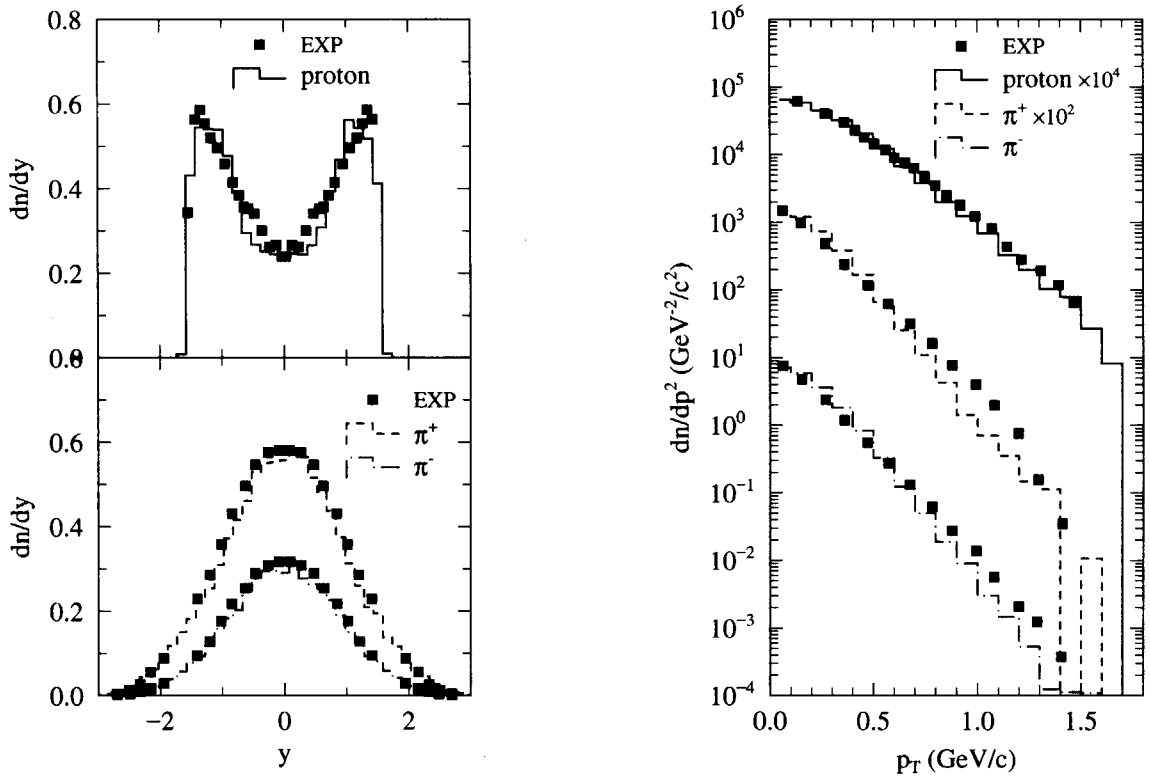

Figure 9. The rapidity y distributions (left panel) and the transverse momentum distributions (right panel) of proton, π^+ and π^- in pp collisions at 12 GeV/c. Histograms are the results obtained from JAM, while the data are from Ref. [22].

Figure 10 shows the energy dependence of the exclusive pion production cross sections in pp reactions. We compare the results of the simulation with the data [23]. Overall agreement is achieved in these exclusive pion productions. Smooth transition from the resonance picture to the string picture at $E_{cm} = 3 \sim 4$ GeV is realized since no irregularity of the energy dependence appears in the calculated results.

For another example of hadron-hadron cross sections, we plot, in Figure 11, the total and

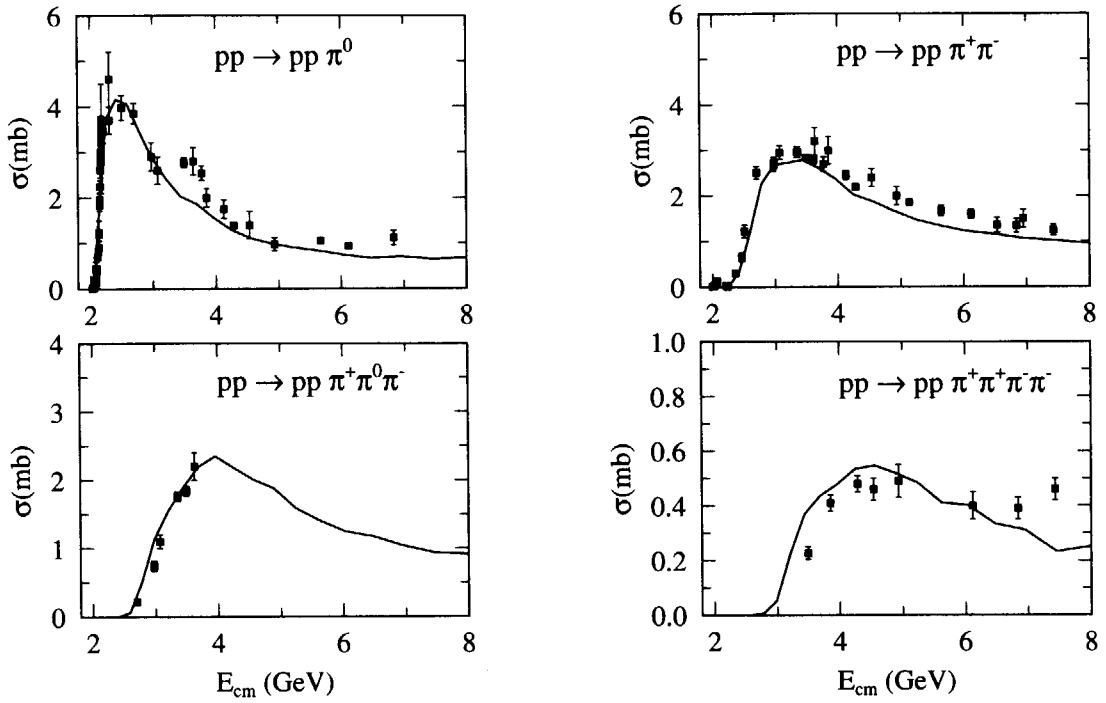

Figure 10. The energy dependence of the exclusive pion production cross sections for proton-proton as a function of c.m. energy. Solid lines are the results obtained from JAM, while the data from Ref. [23].

elastic $\pi^- p$ and $K^+ p$ cross sections parametrized by JAM (upper panel), and the energy dependence of the exclusive cross sections of $K^- p \to \pi^0 \Lambda$ and $K^- n \to \pi^- \Sigma^0$ (lower panel). Data are taken from Refs. [16, 24]

These examples indicates that the parametrization of the elementary hadron-hadron cross sections in JAM is accurate enough for the high energy particle transport calculations.

3.1.3 Comparison with Experimental Data of Thin Target

For validation, we compare the results of the JAM code with experimental data of thin targets. In Figure 12, we plot the invariant transverse mass distribution of proton (left panel), π^- (middle panel), and K^+ (right panel) from proton on thin Au target reaction at 13.7 GeV. The results of JAM (histograms) and data [25] are plotted for each rapidity y bin quoted in the figure. For all ejectiles, the results of JAM agree well with the experimental data [25]. The agreements are also shown in the other targets of Be, Al, and Cu in Ref. [13].

Next comparisons are shown for the reactions of the incident energies around 3 GeV. Figure 13 shows double differential cross section of neutron from p (3 GeV)+ ^{208}Pb (left panel), and invariant cross section of π^- from p (3.17 GeV) + ^{208}Pb (right panel). The histograms are the results of JAM, while data are taken from Refs.[26, 27]. The results of JAM (left panel) well reproduce the neutron yields from beam energy region down to 1 MeV where the neutrons come

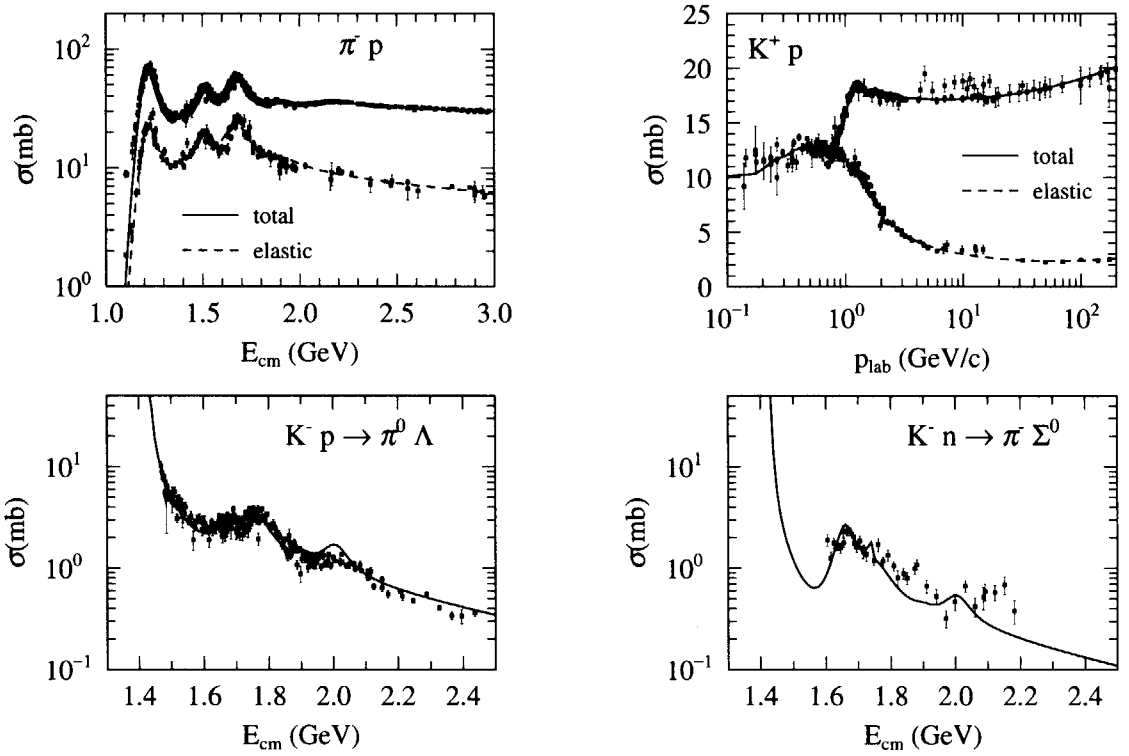

Figure 11. The parametrization of the total and elastic $\pi^- p$ and $K^+ p$ cross sections (upper panel), and the energy dependence of the exclusive cross sections of $K^- p \to \pi^0 \Lambda$ and $K^- n \to \pi^- \Sigma^0$ (lower panel). Data are taken from Refs. [16, 24].

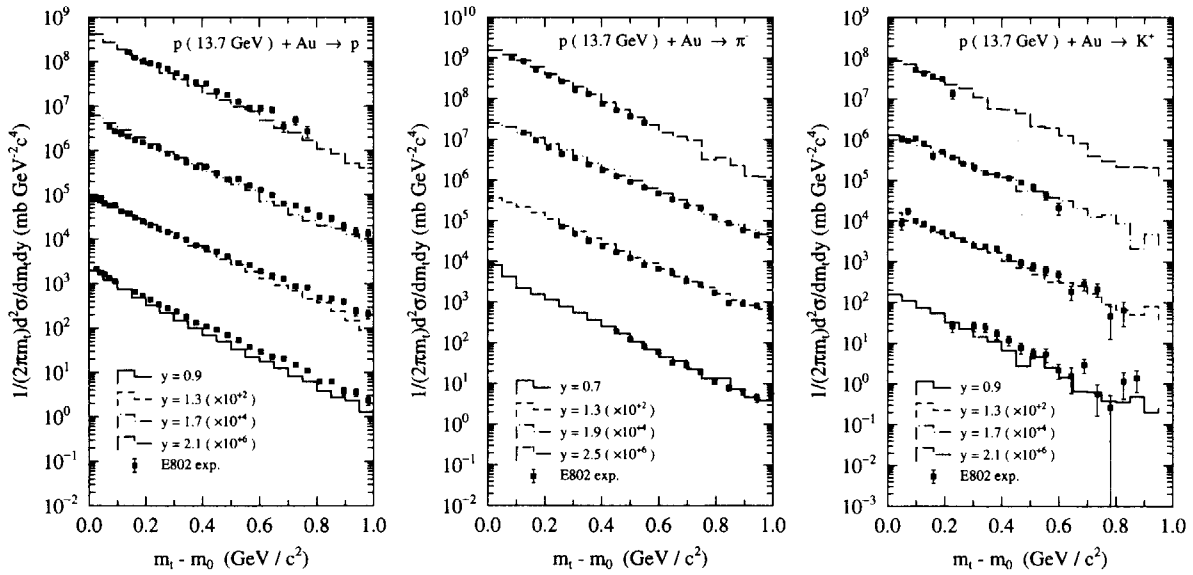

Figure 12. Invariant transverse mass distribution of proton (left panel), π^- (middle panel), and K^+ (right panel) from proton on thin Au target reaction at 13.7 GeV. The results of JAM (histograms) and data [25] are plotted for each rapidity y bin quoted in the figure.

from the evaporation process. In this calculation, statistical decay process is implemented after the intra-nuclear cascade process described by JAM. We have found that the results of JAM are almost comparable to that of Bertini model for the thin target systems at the incident energies below 3.5 GeV.

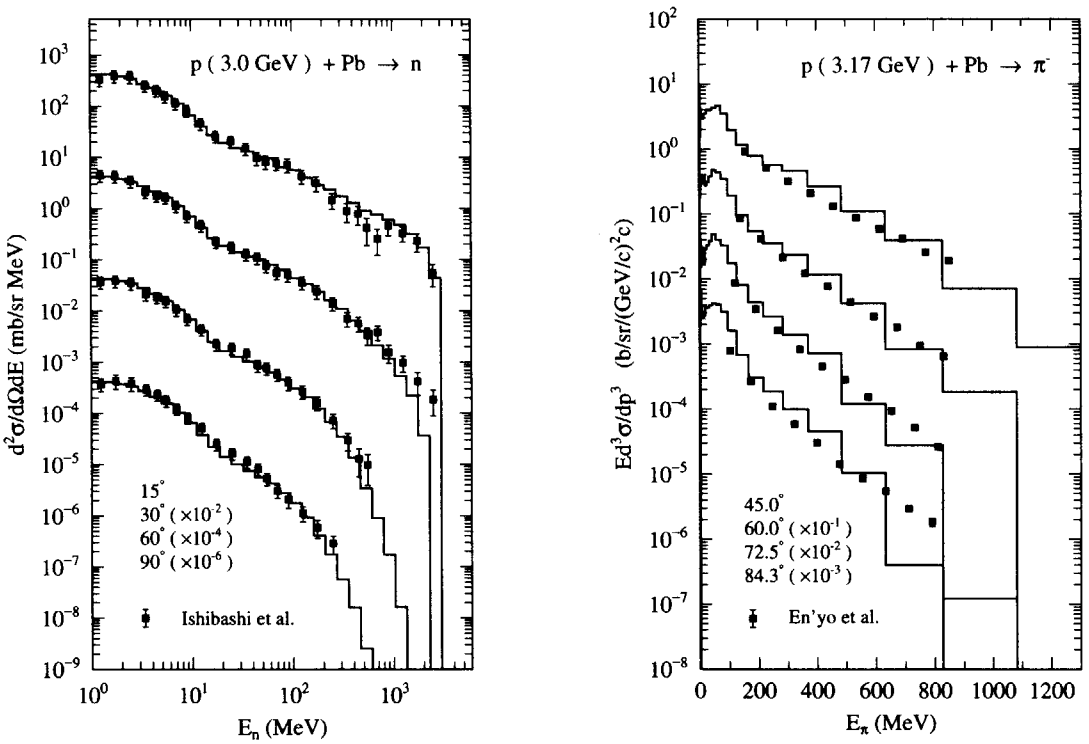

Figure 13. Double differential cross section of neutron from p (3 GeV)+ ^{208}Pb (left panel), and invariant cross section of π^- from p (3.17 GeV) + ^{208}Pb (right panel). The histograms are the results of JAM, while data are taken from Refs.[26, 27].

3.2. Bertini model

The Bertini model [28] treats the interaction between an incident particle and an intranuclear nucleon as a collision between free particles in the degenerated Fermi sea. The version of MECC-3 [3] is installed in NMTC/JAM in which slight modification was made. The target nucleus is divided into three regions having the relative nucleon densities of 0.9, 0.2 and 0.01 to the nucleon density at the center of the nucleus. The Fermi energy of nucleons in each region is supplied by the parametrized data. The constant value of 7 MeV is used as the binding energy of the intranuclear nucleons in each region.

The nucleon-nucleon cross sections are employed to calculate the mean free path of a travelling nucleon and collision probability. We have three options for the nucleon-nucleon cross section data, that in free space, in-medium cross sections of old Cugnon parametrization [29] and new one [29]. These cross sections are shown in Figure 14.

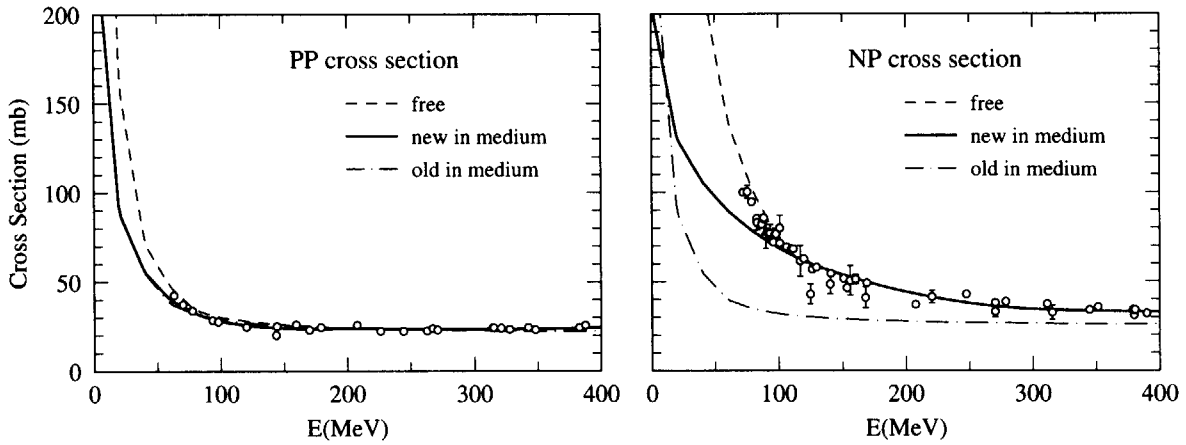

Figure 14. PP (left panel) and NP (right panel) cross sections as a function of lab. energy. The dashed lines denote the free cross sections, the solid line the in medium cross sections of new Cugnon parametrization, while the dash-dotted lines the old one.

The production of pions via an excited state of a nucleon (Δ) is taken into account using the isobar model proposed by Lindenbaum and Steinheimer [30]. The Δ particle is assumed to decay into a nucleon and one or two pions immediately in the Bertini model. The highest inelastic channel is, therefore, the two pion production channel. This is the upper limit of the Bertini model. The exclusion principle is applied by checking whether or not the kinetic energy of a struck particle reaches the cutoff energy which is set to the half height of Coulomb barrier above Fermi level. Reflection and refraction of a particle on the surface of the target nucleus and the boundary between the neighboring regions in the nucleus is not considered.

3.3. ISOBAR code

In NMTC/JAM, the ISOBAR code [31] is employed as an alternative option for the intranuclear cascade calculation. The code is an extended version of VEGAS [31] to supplement the Δ particle capture, the Δ particle-nucleon exchange and so on to the treatment of the Δ particle-nucleon interactions [32].

In the ISOBAR code, the radius of a target nucleus, r, is defined as $r = 1.07 \times A^{1/3} + 2.5$ fm. The target is divided radially into 8 regions. The thickness of the outmost (1st) region is 1.0 fm. The next 5 regions have the thickness of 0.6 fm, and the 7th region has again that of 1.0 fm. Each region has the nucleon density, ρ_i, which corresponds to the fractions of 0.025, 0.1, 0.3, 0.5, 0.7, 0.9 and 0.975 to the nucleon density of the central (8th) region, respectively, where i denotes the region number. The Fermi energy for neutron and proton in the region i is given by the local Fermi momentum obtained by the local Thomas-Fermi approximation. When a travelling particle crosses a boundary of two neighboring regions, the change of its direction of motion by the reflection and refraction at the boundary is calculated according to the Snell's

law [33].

The parametrized in-medium NN cross sections similar to those of Cugnon [29] have been employed [34] instead of the free nucleon-nucleon ones to calculate the mean free path and collision probability of nucleon in a target nucleus.

Moreover, the ISOBAR code takes account of the threshold energies for (p, n) and (n, p) reactions to treat the quasi-elastic collision properly. The threshold energy is added to the cutoff energy which is the sum of the Fermi energy, the binding energy and the Coulomb barrier, and is used to check the escape possibility of a travelling nucleon after colliding in the target.

3.4. SDM model

SDM (Statistical Decay Model) [35] is a simple model of light particle evaporation. We consider only n, p, d, t, ^{3}He, and α evaporation. The emission probability P_x of these particles x is given with the Fermi gas model as,

$$P_x = (2J_x + 1)\, m_x\, \epsilon\, \sigma_x(\epsilon)\, \rho(E)\, d\epsilon, \tag{1}$$

where J_x, m_x, and ϵ are the spin, mass and kinetic energy of the particle x, while $\sigma_x(\epsilon)$ and $\rho(E)$ denote the inverse cross section for the absorption of the particle with energy ϵ and the level density of the residual nucleus with the excitation energy E, respectively. We use the following simple form for $\rho(E)$,

$$\rho(E) = w_0 \exp\left(2\sqrt{a\,E}\right), \tag{2}$$

with $a = A/8$ MeV^{-1} and w_0 is a constant. The inverse cross section is assumed to have the form

$$\sigma_x(\epsilon) = \begin{cases} (1 - U_x/\epsilon)\ \pi R^2 & : \epsilon > U_x \\ 0 & : \epsilon \leq U_x \end{cases} \tag{3}$$

where R denotes the absorption radius and U_x is a Coulomb barrier for the particle x, for which we employ empirical values used in the existing statistical decay code [36]. The excitation energy E in Eq. (2) is given by

$$E = E_0 - \epsilon - Q, \tag{4}$$

where E_0 denotes the excitation energy of the parent nucleus and Q is the reaction Q-value calculated from the simple mass formula. The total emission probability R_x of the particle x is obtained by integrating the available energy of Eq. (1) as

$$\begin{aligned} R_x &= (2J_x + 1)\, m_x \\ &\quad \times \int_{U_x}^{E_0 - Q_x} \epsilon\, \sigma_x(\epsilon)\, \rho(E_0 - Q_x - \epsilon)\, d\epsilon. \end{aligned} \tag{5}$$

This integration can be calculated analytically and the energy spectrum of the emitted particles is given by

$$N(\epsilon_x)\, d\epsilon_x = \frac{\epsilon_x - U_x}{T_x^2}\ \exp\left\{-\frac{\epsilon_x - U_x}{T_x}\right\}\ d\epsilon_x, \tag{6}$$

with

$$a\,T_x^2 = E_0 - U_x - Q_x. \tag{7}$$

In this formulation, we do not consider the γ decay nor the angular momentum dependence. We simulate the whole statistical decay process as a sequential light particle evaporation discussed above by making use of the Monte-Carlo method until no more particle can be emitted. The high energy fission model proposed by Nakahara [5] is implemented in SDM.

3.5. GEM model

The GEM code developed by Furihata [14] is a simulation program of the generalized evaporation and fission model.

3.5.1 Decay width

The generalized evaporation model is based on the Weisskopf-Ewing's formulation. According to the formulation, the decay probability P_j for the emission of a particle j from a parent nucleus i with total kinetic energy in the center-of-mass system between ϵ and $\epsilon + \mathrm{d}\epsilon$ is

$$P_j(\epsilon)d\epsilon = g_j\sigma_{inv}(\epsilon)\frac{\rho_d(E-Q-\epsilon)}{\rho_i(E)}\epsilon d\epsilon, \tag{8}$$

where E (MeV) is excitation energy of a parent nucleus i with mass A_i charge Z_i, and d denotes a daughter nucleus with mass A_d and charge Z_d produced after the emission of ejectile j with mass A_j and charge Z_j in its ground state; σ_{inv} is the cross section for the inverse reaction, ρ_i and ρ_d are level densities (MeV^{-1}) of the parent and the daughter nucleus, respectively; g_j is expressed as $g_j = (2S_j+1)m_j/\pi^2\hbar^2$, with the spin S_j and the mass m_j of the emitted particle j; the Q-value is calculated using the excess mass $M(A,Z)$ as $Q = M(A_j,Z_j) + M(A_d,Z_d) - M(A_i,Z_i)$. In GEM, four mass tables are used to calculate Q-values, according to the following priority: (1) Audi-Wapstra's mass table [37]; (2) theoretical masses calculated by Moller et al. [38]; (3) theoretical masses calculated by Comay et al. [39]; (4) the excess mass calculated by using Cameron's formula.

The cross section for the inverse reaction σ_{inv} is expressed as

$$\sigma_{inv}(\epsilon) = \begin{cases} \sigma_g c_n\,(1+b/\epsilon) & \text{for neutrons} \\ \sigma_g c_j\,(1-V/\epsilon) & \text{for charged particles} \end{cases} \equiv \sigma_g\alpha\left(1+\frac{\beta}{\epsilon}\right), \tag{9}$$

where $\sigma_g = \pi {R_b}^2$ (fm^2) is the geometric cross section, and $V = k_j Z_j Z_d e^2/R_c$ (MeV) is the Coulomb barrier.

The total decay width Γ_j can be calculated by integrating Eq. (8) with respect to the total kinetic energy ϵ from the Coulomb barrier V up to the maximum possible value $(E-Q)$. By using Eq. (9) for σ_{inv}, the total decay width for the particle emission is expressed as

$$\Gamma_j = \frac{g_j\sigma_g\alpha}{\rho_i(E)}\int_V^{E-Q}\epsilon\left(1+\frac{\beta}{\epsilon}\right)\rho_d(E-Q-\epsilon)d\epsilon. \tag{10}$$

According to the Fermi-gas model, the total level density $\rho(E)$ is given by the expression [40]

$$\rho(E) = \begin{cases} \dfrac{\pi}{12}\dfrac{e^{2\sqrt{a(E-\delta)}}}{a^{1/4}(E-\delta)^{5/4}} & \text{for } E \geq E_x \\ \dfrac{\pi}{12}\dfrac{1}{T}e^{(E-E_0)/T} & \text{for } E < E_x \end{cases}, \tag{11}$$

where a (MeV^{-1}) is the level density parameter, and δ (MeV) is the pairing energy of the daughter nucleus. The pairing energy δ is expressed as a sum of separate contributions from neutrons and protons, i.e. $\delta = P(Z_d) + P(N_d)$. The nuclear temperature T is given by $1/T = \sqrt{a/U_x} - 1.5/U_x$, and E_0 is defined as $E_0 = E_x - T(\log T - 0.25\log a - 1.25\log U_x + 2\sqrt{aU_x})$.

3.5.2 Kinetic energy

The probability distribution of the total kinetic energy ϵ of an ejectile and a daughter nucleus is given by Eq. (8). Substituting Eq. (11) into Eq. (8), and normalizing yields

$$P_j(\epsilon) = \begin{cases} g_j\sigma_g\alpha(\epsilon+\beta)\dfrac{e^{(E-Q-\epsilon-E_0)/T}}{T}\dfrac{1}{\Gamma_j} & E-Q-E_x \leq \epsilon \leq E-Q \\ g_j\sigma_g\alpha(\epsilon+\beta)\dfrac{e^{2\sqrt{a(E-Q-\delta-\epsilon)}}}{a^{1/4}(E-Q-\delta-\epsilon)^{5/4}}\dfrac{1}{\Gamma_j} & kV \leq \epsilon \leq E-Q-E_x \end{cases}. \tag{12}$$

The total kinetic energy ϵ is randomly chosen with respect to the probability given by Eq. (12). The direction of the motion $\boldsymbol{w}$ is randomly selected to be isotropic in the center of mass system. Then the excitation energy of the daughter nucleus is calculated as $E_d^* = E - Q - \epsilon$.

The kinetic energy of the ejectile in the laboratory system is calculated in the following manner: The momentum of the ejectile in the center of mass system $\boldsymbol{p_{cm}}$ is calculated by ϵ and the unit vector $\boldsymbol{w}$ of the direction of motion as

$$\boldsymbol{p_{cm}} = \boldsymbol{w}\sqrt{\epsilon^2 + \frac{2m_j m_d \epsilon}{m_j + m_d}}, \tag{13}$$

where m_j and m_d are the masses in MeV of the ejectile and the daughter nucleus, respectively. The momentum $\boldsymbol{p_j}'$ of the ejectile in the laboratory system are obtained by the Lorentz transformation as

$$\begin{aligned} \boldsymbol{p_j}' &= \boldsymbol{p_{cm}} + T_j\boldsymbol{V_i}, \\ \boldsymbol{p_d}' &= -\boldsymbol{p_{cm}} + T_d\boldsymbol{V_i}, \\ T_\ell &= \gamma\left\{\frac{\gamma}{\gamma+1}(\boldsymbol{p_{cm}}\cdot\boldsymbol{V_i}) + E_\ell\right\} \qquad \ell = j \text{ or } d, \end{aligned}$$

where $E_\ell = \sqrt{\boldsymbol{p_{cm}}^2 + m_\ell^2}$ ($\ell = j$ or d), $\gamma = 1/\sqrt{1-\boldsymbol{V_i}^2}$, and the velocity of the parent nucleus in the laboratory system $\boldsymbol{V_i} = \boldsymbol{u}\sqrt{E_r^2 + 2m_i E_r}/(E_r + m_i)$ with the recoil energy E_r of the parent nucleus and the direction of $\boldsymbol{u}$ (unit vector) in the laboratory system after the initial non-equilibrium stage. The kinetic energy in the laboratory system is calculated as $E_\ell' = \sqrt{\boldsymbol{p_\ell}'^2 + m_\ell^2}$ ($\ell = j$ or d).

3.5.3 Ejectiles

In GEM, 66 nuclides, not only in their ground states but also in their excited states, are treated as ejectiles. The selected ejectiles listed in Table 2 satisfy the following criteria: (1) isotopes with $Z_j \leq 12$; (2) naturally existing isotopes or isotopes near the stability line; (3) isotopes with half-life longer than 1 ms.

If the mean lifetime of a resonance is longer than the decay width of the resonance emission, such a resonance can survive during the evaporation process. The excited state is included if its half lifetime $T_{1/2}$ (sec) satisfies the following condition:

$$\frac{T_{1/2}}{\ln 2} > \frac{\hbar}{\Gamma_j^*}, \tag{14}$$

where Γ_j^* is the decay width of the resonance emission. Γ_j^* can be calculated in the same manner as for a ground state particle emission. The Q-value for the resonance emission is expressed as $Q^* = Q + E_j^*$, where E_j^* is the excitation energy of the resonance. The spin state of the resonance S_j^* is used in the calculation of g_j, instead of the spin of the ground state S_j. Although the mass of an excited state is different from that of the ground state, the ground state mass m_j is used, because the difference of the masses is negligible.

Instead of treating a resonance as an independent particle, the decay width of the ground state particle emission is simply enhanced. The decay width Γ_j is redefined as

$$\Gamma_j = \Gamma_j^0 + \sum_n \Gamma_j^n, \tag{15}$$

where Γ_j^0 is the decay width of the ground state particle j emission, and Γ_j^n is that of the nth excited state of the particle j emission which satisfies Eq. (14).

The total kinetic energy distribution of the excited particle emission is assumed to be the same as that of the ground state particle emission. S_j^*, E_j^*, and $T_{1/2}$ used in this study were extracted from the ENSDF database maintained by the National Nuclear Data Center [41].

The other details of the parameters of Coulomb barrier, the inverse cross sections, the level density and the high energy fission model are discussed in Ref. [14].

Table 2. Ejectiles of GEM model.

ID	nuclide	Z	A	spin	ID	nuclide	Z	A	spin
1	n	0	1	1/2	36	^{14}O	8	14	0
2	p	1	1	1/2	37	^{15}O	8	15	1/2
3	d	1	2	1	38	^{16}O	8	16	0
4	t	1	3	1/2	39	^{17}O	8	17	5/2
5	^{3}He	2	3	1/2	40	^{18}O	8	18	0
6	α	2	4	0	41	^{19}O	8	19	5/2
7	^{6}He	2	6	0	42	^{20}O	8	20	0
8	^{8}He	2	8	0	43	^{17}F	9	17	5/2
9	^{6}Li	3	6	1	44	^{18}F	9	18	1
10	^{7}Li	3	7	3/2	45	^{19}F	9	19	1/2
11	^{8}Li	3	8	2	46	^{20}F	9	20	2
12	^{9}Li	3	9	3/2	47	^{21}F	9	21	5/2
13	^{7}Be	4	7	3/2	48	^{18}Ne	10	18	0
14	^{9}Be	4	9	3/2	49	^{19}Ne	10	19	1/2
15	^{10}Be	4	10	0	50	^{20}Ne	10	20	0
16	^{11}Be	4	11	1/2	51	^{21}Ne	10	21	3/2
17	^{12}Be	4	12	0	52	^{22}Ne	10	22	0
18	^{8}B	5	8	2	53	^{23}Ne	10	23	5/2
19	^{10}B	5	10	3	54	^{24}Ne	10	24	0
20	^{11}B	5	11	3/2	55	^{21}Na	11	21	3/2
21	^{12}B	5	12	1	56	^{22}Na	11	22	3
22	^{13}B	5	13	3/2	57	^{23}Na	11	23	3/2
23	^{10}C	6	10	0	58	^{24}Na	11	24	4
24	^{11}C	6	11	3/2	59	^{25}Na	11	25	5/2
25	^{12}C	6	12	0	60	^{22}Mg	12	22	0
26	^{13}C	6	13	1/2	61	^{23}Mg	12	23	3/2
27	^{14}C	6	14	0	62	^{24}Mg	12	24	0
28	^{15}C	6	15	1/2	63	^{25}Mg	12	25	5/2
29	^{16}C	6	16	0	64	^{26}Mg	12	26	0
30	^{12}N	7	12	1	65	^{27}Mg	12	27	1/2
31	^{13}N	7	13	1/2	66	^{28}Mg	12	28	0
32	^{14}N	7	14	1					
33	^{15}N	7	15	1/2					
34	^{16}N	7	16	2					
35	^{17}N	7	17	1/2					

3.5.4 Comparison with Experimental Data of Isotope Production

We have implemented the GEM model into NMTC/JAM. For validation, we compare the result of the nuclear reaction calculation by JAM with GEM with experimental data. In Figure 15, we plot the isotope production cross sections of He, Li and Be from the Ag(p,x) reaction at 480 MeV. The experimental data are taken from Ref. [42]. The results of JAM with GEM well reproduce the experimental data. The isotopes of Li and Be are hardly reproduced by the other model, SDM or DRES.

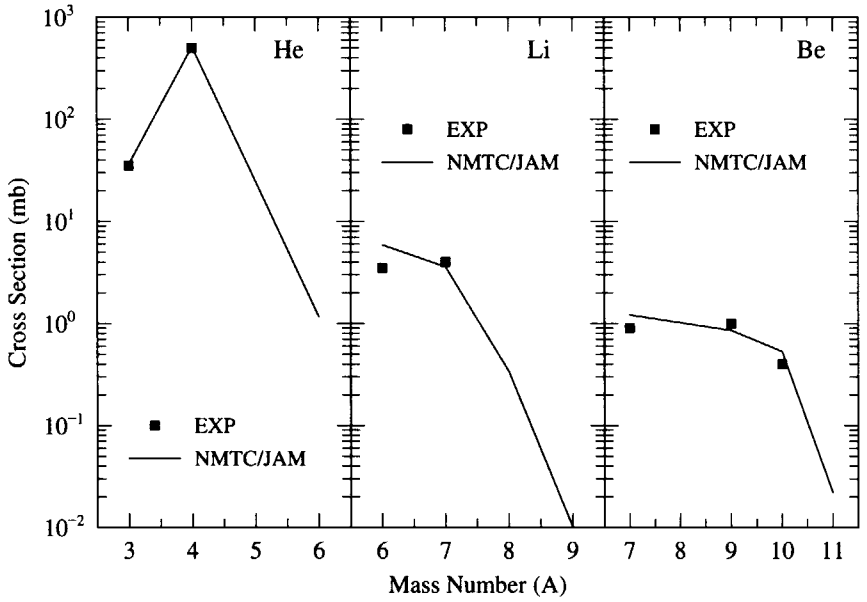

Figure 15. The isotope production cross sections of He, Li and Be from the Ag(p,x) reaction at 480 MeV. The experimental data are taken from Ref. [42]. The results of JAM with GEM are shown by the solid lines.

3.6. DRES model

The original version of DRES was developed by Dresner as EVAP [4]. NMTC/JAM implements a slightly modified version from EVAP-4 [43]. The emissions of six particles of n, p, d, t, ^{3}He, and α are taken into account. The formula of DRES is almost the same as in SDM, but it takes into account the pairing energy and the mass dependent level density parameter. Then the expression of the level density Eq. (2) is given by

$$\rho(E) = w_0 \exp\left(2\sqrt{a\,(E-\delta)}\right), \tag{16}$$

with

$$a = \frac{A}{B}\left(1 + Y\frac{\Delta^2}{A^2}\right), \tag{17}$$

where δ is the pairing energy and a is the level density parameter, in which B is treated as a parameter in the code. Here, Δ is given as $A-2Z$ by the use of the mass number and the atomic number of a residual nucleus. A value of 1.5 is given to Y. As far as the level density parameter is concerned, the following three options are available in NMTC/JAM: mass dependent value of A/8, the level density parameter derived by Baba [44] and that derived by Ignatyuk [45] with the parameters proposed by Mengoni et al. [46]. The Ignatyuk formula takes account of the excitation energy dependence in the level density.

3.7. High Energy Fission Model

The high energy fission process is treated as a competitive process with the evaporation of SDM and DRES. Before going to the evaporation stage, the fission reaction is selected if a random number exceeds the fission probability P_f represented with the fission width Γ_f and the neutron emission width Γ_n

$$P_f = \frac{1}{1+\Gamma_n/\Gamma_f}. \tag{18}$$

According to a statistical model, the ratio of the neutron emission width to the fission one is given by the following equation,

$$\frac{\Gamma_n}{\Gamma_f} = \frac{4A^{2/3}a_f(U-Q_n)}{K_0 a_n\left(2\sqrt{a_f(U-E_f)}-1\right)} \cdot \exp\left(2\sqrt{a_n(U-E_n)} - 2\sqrt{a_n(U-E_n)}\right) \tag{19}$$

with

$$K_0 = h^2/(8p^2 m r_0^2) \tag{20}$$

where Q_n the neutron binding energy, E_f the fission barrier height, a_n the level density parameter for neutron emission, a_f the one for fission, m the neutron mass and r_0 the nucleus radius. The ratio of the level density parameter a_f/a_n is determined by an expression fitted to the fission cross section data obtained by Il'inov et al. [47] for the incident proton energies of 150, 660 and 1,000 MeV. The data of Il'inov and those of Kupryanov et al. [48] are employed to determine

the fission barriers for the nuclides with mass number $A \leq 224$ and for the ones $A \geq 225$, respectively. The mass distribution of fission fragments for sub-actinide nuclides is calculated using a Gaussian distribution with the half width derived by Neuzil and Fairhal [49]. For actinide nuclides, it is calculated with the folded Gaussian distribution with the parameters obtained by fitting to the data of Grass et al. [50] for the α-induced fission of ^{238}Pu. The charge of the fission fragment is determined by a Gaussian distribution with the parameters obtained by Pik-Pichak and Strutinskii [51] for the masses of given fragments. The detailed process to determine the residual fragments are reported in Ref. [4].

3.8. Preequilibrium Calculation Model

Between the cascade process and the evaporation process, one can use the preequilibrium model for the fast emitted particles. For the preequilibrium calculation, the closed form exciton model proposed by Gudima et al. [52] is adopted. The number of particles in the exciton state, p_0, at an initial stage of the preequilibrium process is the same as that of allowed collisions during the intranuclear cascade process. The number of holes, h_0, is $p_0 - 1$. The preequilibrium calculation is carried out in the framework of a Monte Carlo algorithm formulated by Nakahara and Nishida [53].

3.9. Nucleon-Nucleus Cross Sections

It is important to use reliable cross section data of nucleon-nucleus for simulating nucleon and meson transport in medium. In NMTC/JAERI97, Pearlstein's systematics [7] is used for non-elastic and elastic cross sections of neutron-nucleus reactions, while for proton-nucleus reactions, elastic collision is not considered, and non-elastic cross section is determined by the Bertini model [28], which is the intra-nucleus cascade model used in NMTC/JAERI97. We have made new systematics based on Pearlstein's systematics for proton as well as neutron nucleus cross sections.

In Figure 16, we plot the results of the new systematics (bold solid lines) for the non-elastic cross section of p+C and p+Pb. In the same figure, we also show the results of LA150 [54] (gray lines), the original Pearlstein's systematics (dashed lines), and the simulation results of the Bertini model (solid lines up to 3.5 GeV) and the JAM code (solid line above 3.5 GeV). Experimental data are taken from Ref. [55]. The present results well follow the experimental data and also LA150 data, while the result of the original Pearlstein's systematics overestimates the data below 100 MeV. The connection of the simulation results between Bertini and JAM is not smooth and in the low energy region, the results of Bertini are too high. This is one of the reason to create present new systematics.

We have also created new systematics based on the Pearlstein's systematics [7] for nucleon-nucleus elastic angular distributions. Figure 17 shows the results of the new systematics of

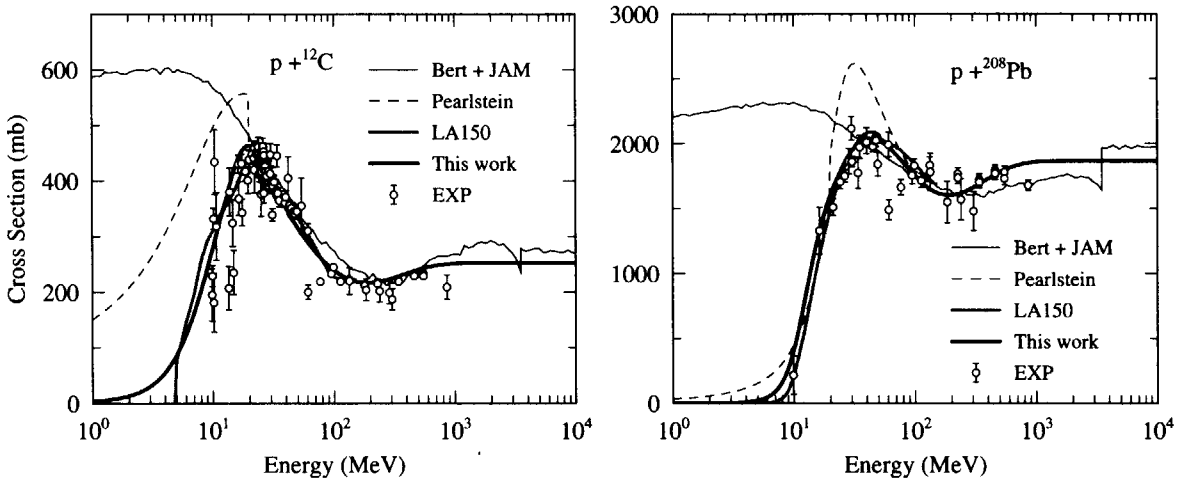

Figure 16. Non-elastic cross section of p+C (left panel) and p+Pb (right panel). The results of the new systematics are denoted by bold solid lines. In the same figure, we also show the results of LA150 [54] (gray lines), the original Pearlstein's systematics (dashed lines), and the simulation results of the Bertini model (solid lines up to 3.5 GeV) and JAM code (solid line above 3.5 GeV). Experimental data are taken from Ref. [55].

the angular distribution of the elastic cross sections. We compare the present results with the experimental data [55] and LA150. The present results well reproduce the experimental data for allover the targets and the incident energies above 150 MeV up to 1 GeV. Even below 150 MeV, the present results have enough accuracy for the transport calculations. Indeed, Figure 18 shows the elastic angular distributions of various targets with 55 MeV (left panel) and 75 MeV (right panel) neutron. Experimental data are taken from Ref. [56]. Though, for some targets, the present results underestimate the experimental data at the angular regions above the second peak, the agreement with the data is very nice at the region of the first peak.

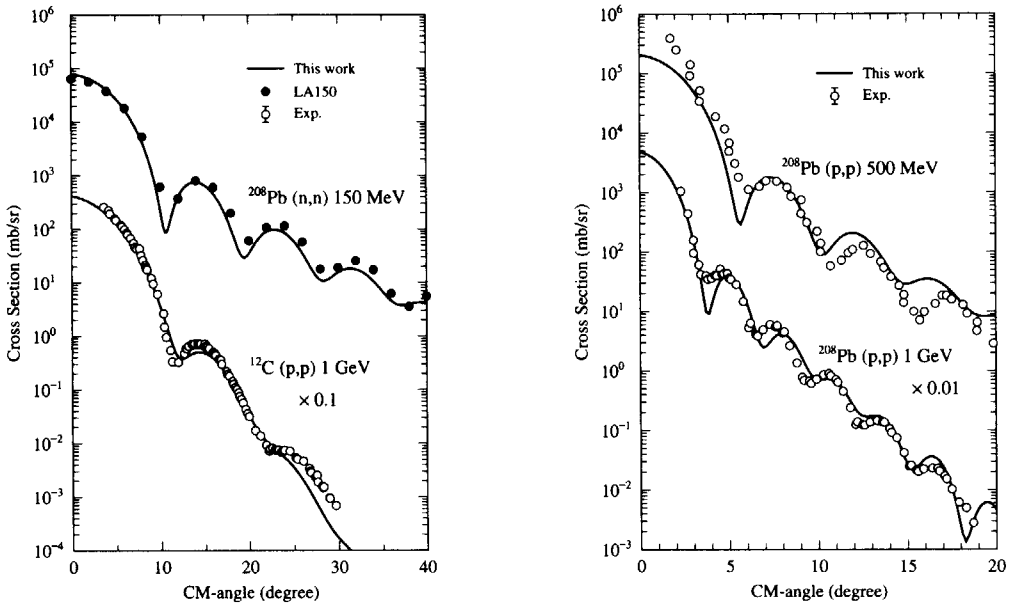

Figure 17. Angular distribution of the elastic cross sections. The results of the new systematic is denoted by solid lines.

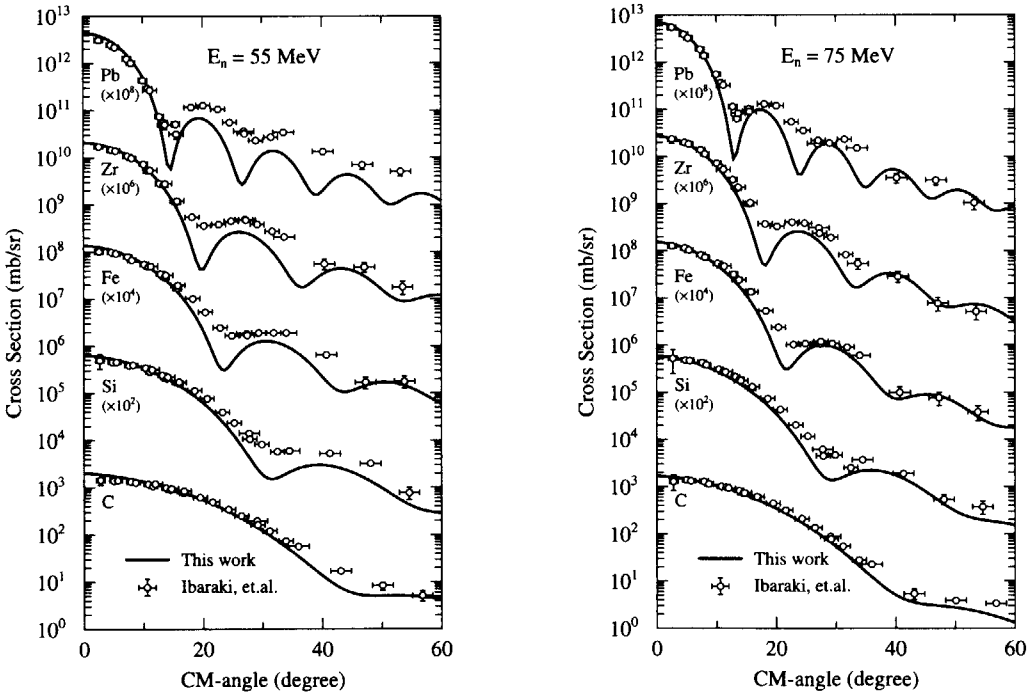

Figure 18. Angular distribution of the elastic cross sections. The results of the new systematic is denoted by solid lines. The incident energy of neutron is 55 MeV (left panel), and 75 MeV (right panel). Targets are ^{208}Pb, ^{90}Zr, ^{56}Fe, ^{28}Si and ^{12}C from upward, respectively.

3.10. Comparison with Experimental Data of Thick Target

We have applied NMTC/JAM to thick target analysis. There do not exist so many reliable experimental data of neutron yields from thick targets with proton beam above 3 GeV. We have here picked up two experiments carried out at KEK and AGS for the comparison with NMTC/JAM calculations.

We have first applied NMTC/JAM to the neutron production from lead targets with 12 GeV protons. The experiment has been done in KEK by means of the Mn-bath moderation method with 20 and 10 cm diameter and 60 cm long cylindrical lead target [57]. Measurements were performed while changing the length of the target in steps of 10 cm in order to obtain the target length dependence of the neutron yields. Calculations are also done tracing the experimental conditions by NMTC/JAM combined with MCNP4A for the neutron below 20 MeV. Figure 19 shows the neutron yields as a unit of the number of neutrons per an incident proton (n/p) as a function of the target length. The open circles (10 cm diameter case) and solid circles (20 cm diameter case) with error bars are data taken from Ref. [57], the solid lines are the results of NMTC/JAM. The results of NMTC/JAM reproduced quit well the experimental data.

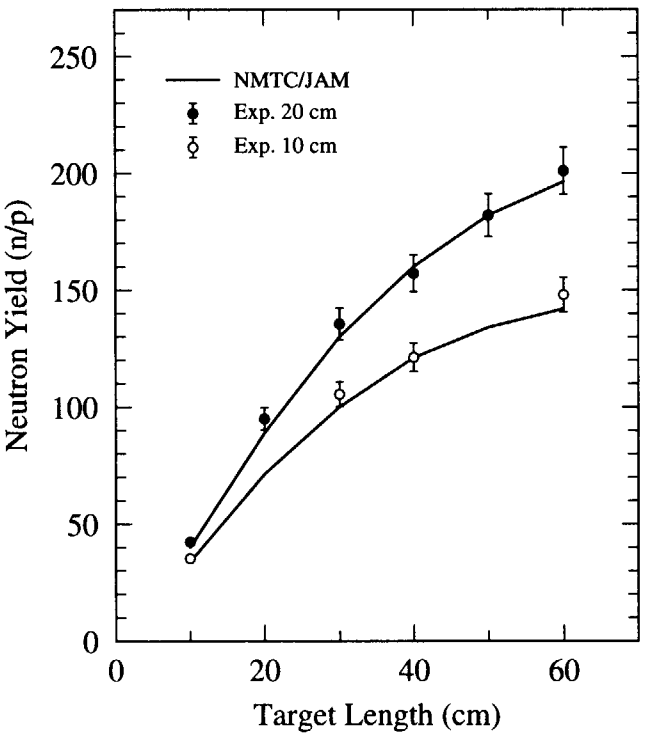

Figure 19. The neutron yields from lead target with 12 GeV protons as a unit of the number of neutrons per an incident proton (n/p) as a function of the target length. The open circles (10 cm diameter case) and solid circles (20 cm diameter case) with error bars are data taken from Ref. [57], the solid lines are the results of NMTC/JAM.

Next comparison is done for experiments under the ASTE (AGS Spallation Target Experiment) collaboration [12]. This experiment has been carried out using a thick mercury target, which is a 20 cm diameter and 130 cm long cylinder, and detecting the reaction rate distributions along the cylindrical surface of the target by the activation technique at incident proton energies of 1.6, 12 and 24 GeV. Various activation detectors such as the ^{115}In(n,n')^{115m}In, ^{93}Nb(n,2n)^{92m}Nb, ^{209}Bi(n,xn) reactions with threshold energies ranging from 0.3 to 70.5 MeV were employed to obtain the reaction rate data for estimating spallation source neutron characteristics of the mercury target.

Figure 20 shows the distribution of the ^{27}Al(n,α)^{24}Na (left-hand side) and ^{209}Bi(n,4n)^{206}Bi (right-hand side) reaction rates along the cylindrical surface of the mercury target bombarded with 1.6, 12, and 24 GeV protons. The thresholds of the reactions are 3.3 MeV for the ^{27}Al(n,α)^{24}Na case and 23 MeV for the ^{209}Bi(n,4n)^{206}Bi case, while the most effective energies of neutron for these reactions are about 10 and 30 MeV, respectively. The results of NMTC/JAM, denoted by the solid histograms in these figures, reproduce the experimental distribution quite well for all positions and all energies, though the result at 1.6 GeV for Bi case slightly underestimates the data.

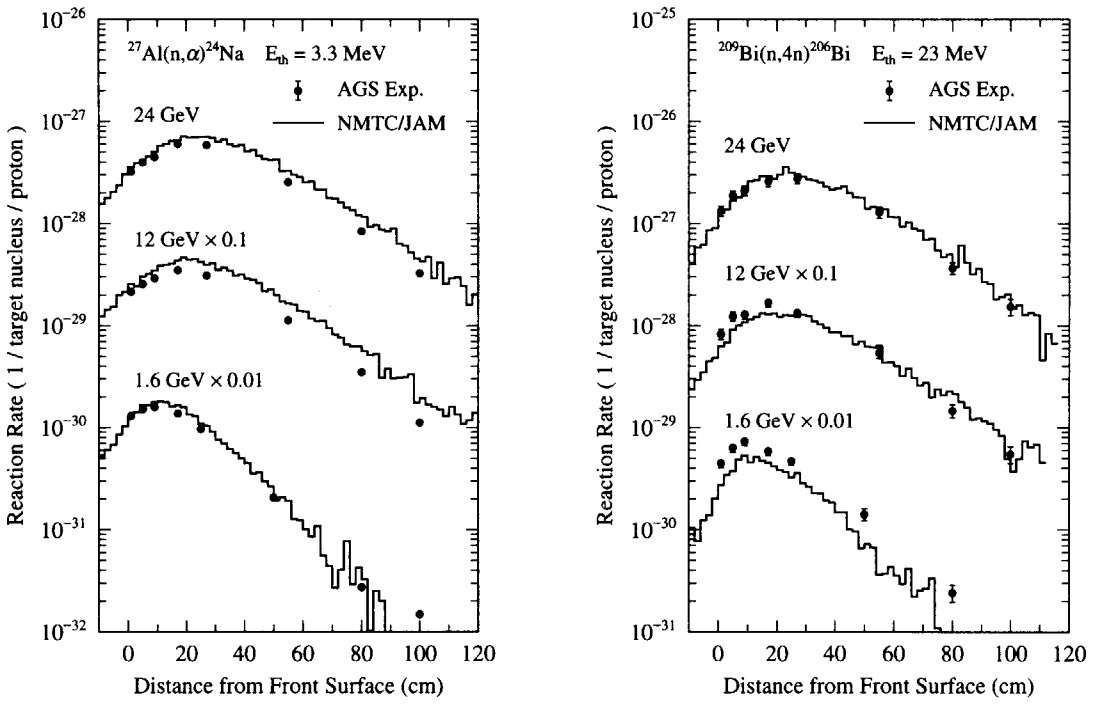

Figure 20. Longitudinal distribution of the ^{27}Al(n,α)^{24}Na (left-hand side) and ^{209}Bi(n,4n)^{206}Bi (right-hand side) reaction rates along the cylindrical surface of the mercury target bombarded with 1.6, 12 and 24 GeV protons. The solid histograms denote the results of NMTC/JAM.

4. Geometry Configuration

The geometry configuration is described with the extended version of Combinatorial Geometry (CG) package, MARS (Multiple Array System) [8, 9] in NMTC/JAM. The concepts of "body" and "region" are introduced in MARS for expressing the geometry configuration. These two concepts are defined as follows:

1) Body is the unit configuration such as rectangular parallelepiped, sphere, right circular cylinder and so on for describing the geometry configuration.

2) Region is the minimum unit of the geometry configuration and is defined by the combination of bodies using the set operator such as unions, differences and intersections. Region is composed of a single material.

4.1. Body Description

The following 13 kinds of basic geometry shapes (bodies) are prepared in the present MARS:

(a) Right Parallelepiped (RPP)
(b) Sphere (SPH)
(c) Right Circular Cylinder (RCC)
(d) Right Elliptical Cylinder (REC)
(e) Truncated Right Angle Cone (TRC)
(f) Ellipsoid (ELL)
(g) Right Angle Wedge (WED)
(h) Box which specifies the vertex and one of the corner (BOX)
(i) Arbitrary Polyhedron (ARB)
(j) Alternative Body Description (BPP or WPP)
(k) General Ellipsoid (GEL)
(l) Truncated Right Elliptical Cone (QUA)
(m) Torus (TOR)

In addition to the above bodies, the following 9 planes and infinite cylinder are available to define the bodies.

(a) Plane defined by Hesse Formula (P)
(b) Plane normal to X-axis (PX)
(c) Plane normal to Y-axis (PY)
(d) Plane normal to Z-axis (PZ)
(e) Plane defined by generalized Hesse Formula (PS)

(f) Cylinder General (C)

(g) Cylinder parallel to X-axis (CX)

(h) Cylinder parallel to Y-axis (CY)

(i) Cylinder parallel to Z-axis (CZ)

The type of body is recognized in the code by the character identification symbol indicated in the above parentheses. The coordinate systems required to define the bodies are summarized in the following Tables 3,4,5.

Table 3. Body configuration data (1).

Body Type	Symbol	input data					
Right Parallelepiped	RPP	X_{min}	X_{max}	Y_{min}	Y_{max}	Z_{min}	Z_{max}
Sphere	SPH	V_x	V_y	V_z	R		
Right Circular Cylinder	RCC	V_x	V_y	V_z	H_x	H_y	H_z
		R					
Right Elliptic Cylinder	REC	V_x	V_y	V_z	H_x	H_y	H_z
		R_{1x}	R_{1y}	R_{1z}	R_{2x}	R_{2y}	R_{2z}
Truncated Right Cone	TRC	V_x	V_y	V_z	H_x	H_y	H_z
		R_1	R_2				
Ellipsoid	ELL	V_{1x}	V_{1y}	V_{1z}	V_{2x}	V_{2y}	V_{2z}
		R					
Right Angle Wedge	WED	V_x	V_y	V_z	H_{1x}	H_{1y}	H_{1z}
		H_{2x}	H_{2y}	H_{2z}	H_{3x}	H_{3y}	H_{3z}
Box	BOX	V_x	V_y	V_z	H_{1x}	H_{1y}	H_{1z}
		H_{2x}	H_{2y}	H_{2z}	H_{3x}	H_{3y}	H_{3z}
Arbitrary Polyhedron	ARB	V_{1x}	V_{1y}	V_{1z}	V_{2x}	V_{2y}	V_{2z}
		V_{3x}	V_{3y}	V_{3z}	V_{4x}	V_{4y}	V_{4z}
		V_{5x}	V_{5y}	V_{5z}	V_{6x}	V_{6y}	V_{6z}
		V_{7x}	V_{7y}	V_{7z}	V_{8x}	V_{8y}	V_{8z}
		1234	*3456*	*5678*	*1278*	*1467*	*2358*

Table 4. Body configuration data (2).

Body Type	Symbol	input data					
Alternative Body	BPP	X_{min}	X_{max}	Y_{min}	Y_{max}	Z_{min}	Z_{max}
		θ_1	θ_2	θ_3			
Alternative Body	WPP	X_{min}	X_{max}	Y_{min}	Y_{max}	Z_{min}	Z_{max}
		θ_1	θ_2	θ_3			
General Ellipsoid	GEL	V_x	V_y	V_z	R_{1x}	R_{1y}	R_{1z}
		R_{2x}	R_{2y}	R_{2z}	R_{3x}	R_{3y}	R_{3z}
Truncated Right Elliptical	QUA	a	b	c	d	e	f
		g	h	Z_1	Z_2		
Torus	TOR	X_0	Y_0	Z_0	R	a	b
		F_{xyz}	θ_1	θ_2			

Table 5. Body configuration data (3).

Body Type	Symbol	input data					
Plane defined by Hesse Formula	P	H_x	H_y	H_z	D		
Plane normal to X-axis	PX	D					
Plane normal to Y-axis	PY	D					
Plane normal to Z-axis	PZ	D					
Plane defined by generalized Hesse Formula	PS	V_x	V_y	V_z	H_x	H_y	H_z
		D					
Cylinder	C	V_x	V_y	V_z	H_x	H_y	H_z
		D					
Cylinder parallel to X-axis	CX	V_x	V_y	V_z	R		
Cylinder parallel to Y-axis	CY	V_x	V_y	V_z	R		
Cylinder parallel to Z-axis	CZ	V_x	V_y	V_z	R		

4.1.1 Right Parallelepiped (RPP)

RPP is defined by the minimum and maximum values of the x, y and z coordinates which bound the parallelepiped.

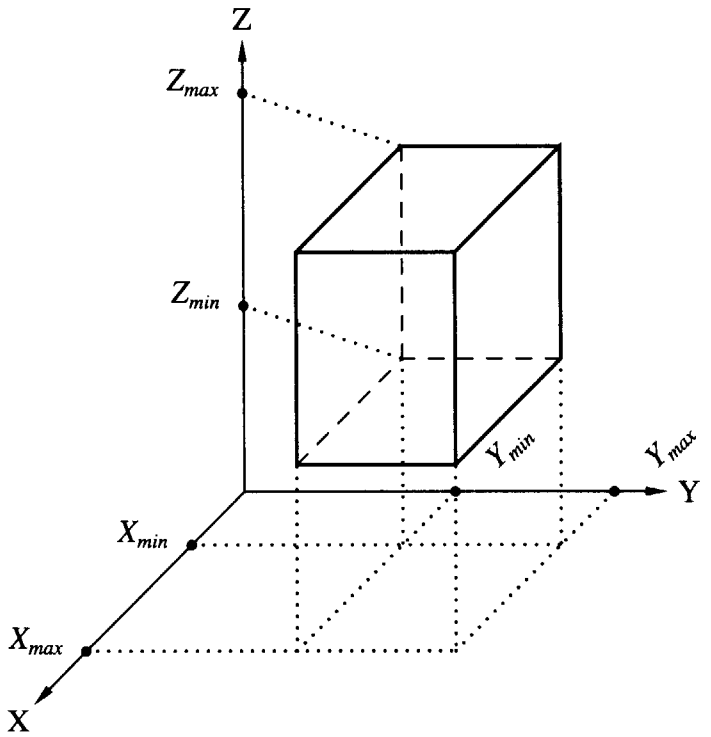

Figure 21. Illustration of a right parallelepiped (RPP).

4.1.2 Sphere (SPH)

SPH is defined by the center with the coordinates (V_x, V_y, V_z) and the radius R.

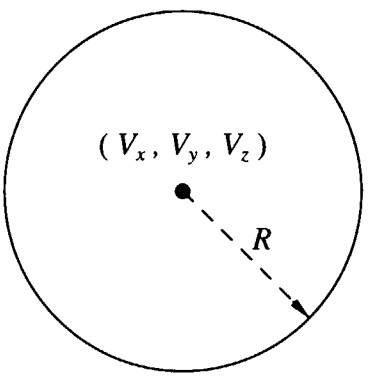

Figure 22. Illustration of a sphere (SPH).

4.1.3 Right Circular Cylinder (RCC)

RCC is defined by the center coordinates (V_x, V_y, V_z) of the vase circle with radius R, and the height vector with the component of (H_x, H_y, H_z).

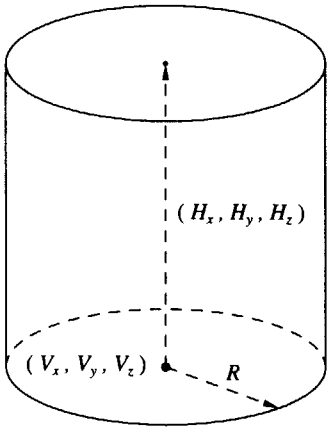

Figure 23. Illustration of a right circular cylinder (RCC).

4.1.4 Right Elliptical Cylinder (REC)

REC is defined by the base ellipse and the height vector. The center coordinates of ellipse (V_x, V_y, V_z), the vector components of the major and minor axes of the ellipse (R_{1x}, R_{1y}, R_{1z}) and (R_{2x}, R_{2y}, R_{2z}), and the component of the height vector (H_x, H_y, H_z).

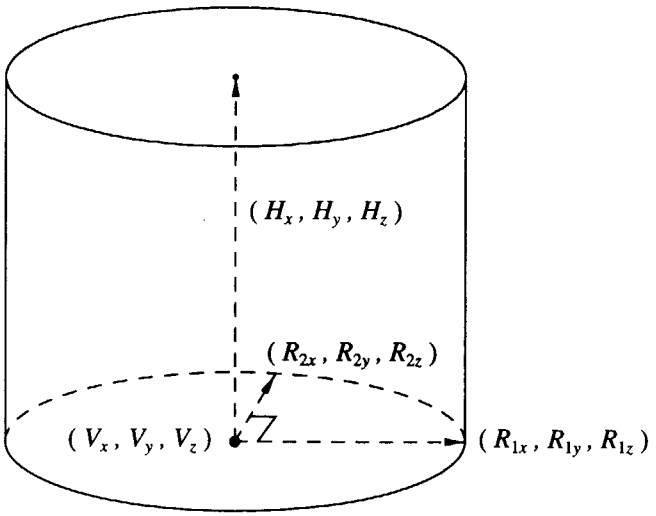

Figure 24. Illustration of a right elliptical cylinder (REC).

4.1.5 Truncated Right Angle Cone (TRC)

TRC is defined by the base and top circles with radii R_1 and R_2 which intersect at right angles with the height vector with the component (H_x, H_y, H_z) originated from the center coordinate of the base circle (V_x, V_y, V_z).

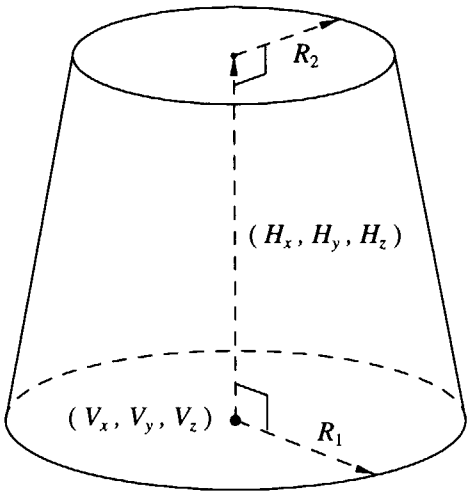

Figure 25. Illustration of a truncated right angle cone (TRC).

4.1.6 Ellipsoid (ELL)

ELL is defined by the two center coordinates of the foci of vertices (V_{1x}, V_{1y}, V_{1z}) and (V_{2x}, V_{2y}, V_{2z}), and the length of the major axis R.

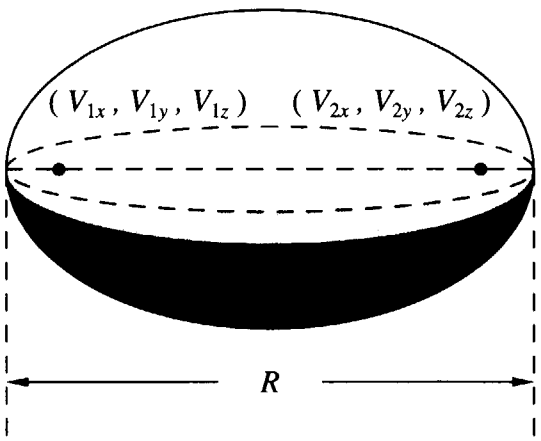

Figure 26. Illustration of a ellipsoid (ELL).

4.1.7 Right Angle Wedge (WED)

WED is defined by one of the corner of the base angle wedge with coordinates (V_x, V_y, V_z) and a set of mutually perpendicular three vectors. These vectors are determined by the three coordinates of (H_{1x}, H_{1y}, H_{1z}), (H_{2x}, H_{2y}, H_{2z}) and (H_{3x}, H_{3y}, H_{3z}).

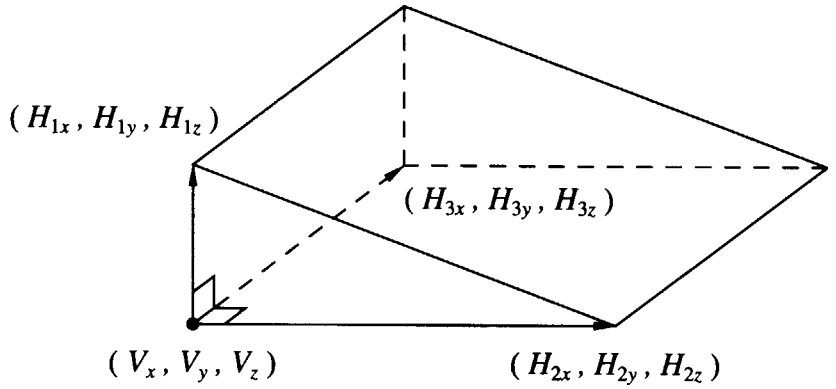

Figure 27. Illustration of a right angle wedge (WED).

4.1.8 Box (BOX)

BOX is defined by the coordinates of one of the corners (V_x, V_y, V_z), and a set of mutually perpendicular three vectors. These vectors are determined by the three coordinates of (H_{1x}, H_{1y}, H_{1z}), (H_{2x}, H_{2y}, H_{2z}) and (H_{3x}, H_{3y}, H_{3z}).

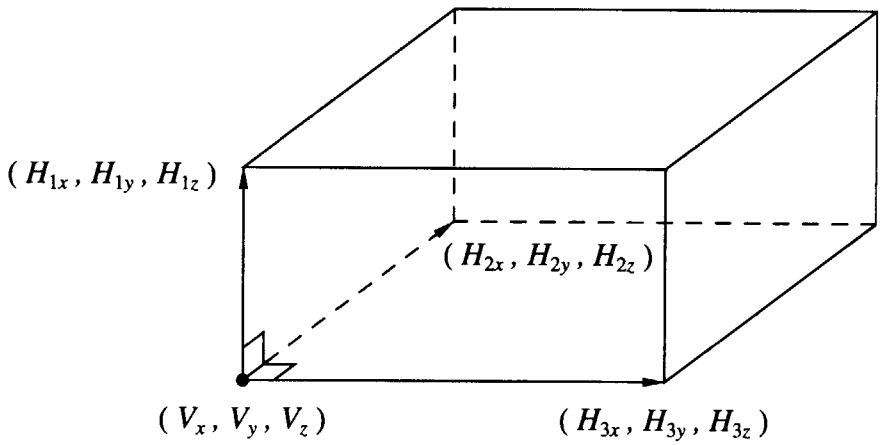

Figure 28. Illustration of a box (BOX).

4.1.9 Arbitrary Polyhedron (ARB)

ARB is defined by the coordinates of eight corners of the polyhedron from (V_{1x}, V_{1y}, V_{1z}) to (V_{8x}, V_{8y}, V_{8z}). Each of the six faces is described by a four-digits integer, representing the four vertex points of the face.

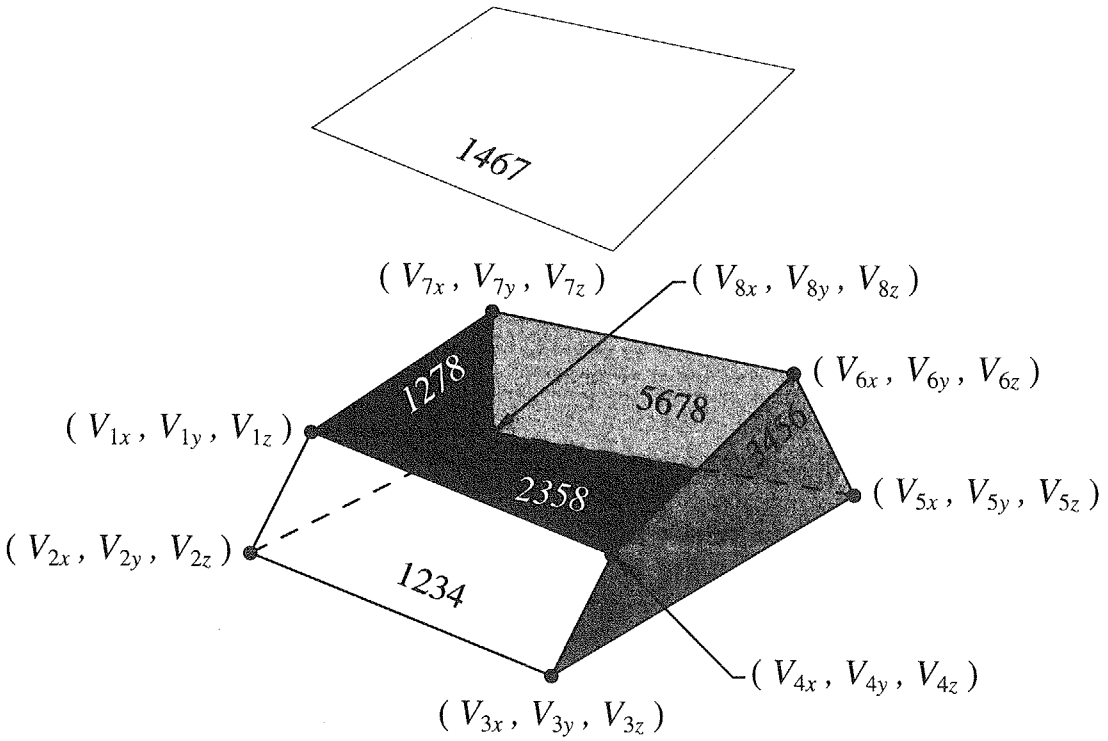

Figure 29. Illustration of an arbitrary polyhedron (ARB).

4.1.10 Alternative Body Description (BPP, WPP)

BPP and WPP are bodies obtained by the rotation of the box and the right angle wedge with respect to the X, Y, and Z axes. BPP is for the rotation of the box and WPP is for the right angle wedge, respectively. BPP and WPP require the user to describe rectangular parallelepiped, similar to RPP, with three rotation angles. The first angle is for the first rotation of RPP which is the X-Y rotation about the Z-axis and is positive in the clockwise direction (1st step). The second angle is a Y-Z rotation about the X-axis and is positive in the clockwise direction (2nd step). The body rotated by the 1st step procedure is again rotated by the second angle in the 2nd step procedure. The third angle is a X-Z rotation about the Y-axis and is positive in the counter clockwise direction. Due to this procedure, the body rotated in the 2nd procedure is again rotated about the Y-axis by the third angle input. The above procedures are illustrated in the following Figures.

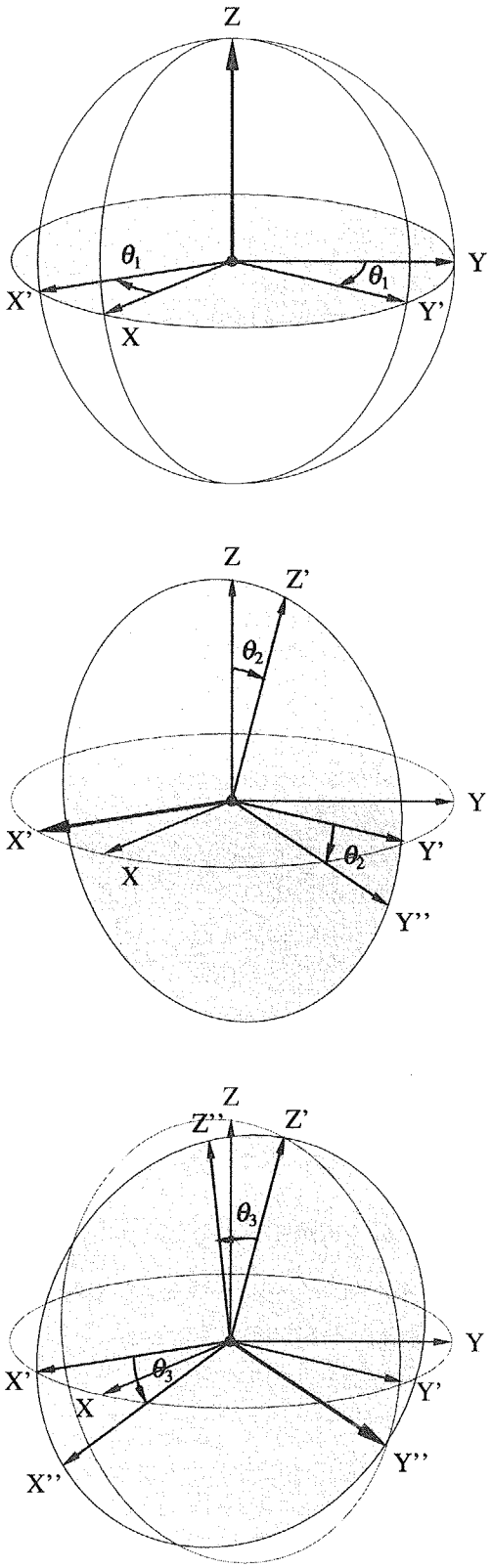

Figure 30. 1st Rotation from (X, Y, Z) to (X', Y', Z) coordinate by angle θ_1 around Z-axis. 2nd Rotation from (X', Y', Z) to (X', Y'', Z') coordinate by angle θ_2 around X'-axis. 3rd Rotation from (X', Y'', Z') to (X'', Y'', Z'') coordinate by angle θ_3 around Y"-axis.

4.1.11 General Ellipsoid (GEL)

GEL is defined by the center coordinates of the ellipsoid (V_x, V_y, V_z), the mutually perpendicular three vectors for the three radii with the components of (R_{1x}, R_{1y}, R_{1z}), (R_{2x}, R_{2y}, R_{2z}) and (R_{3x}, R_{3y}, R_{3z}).

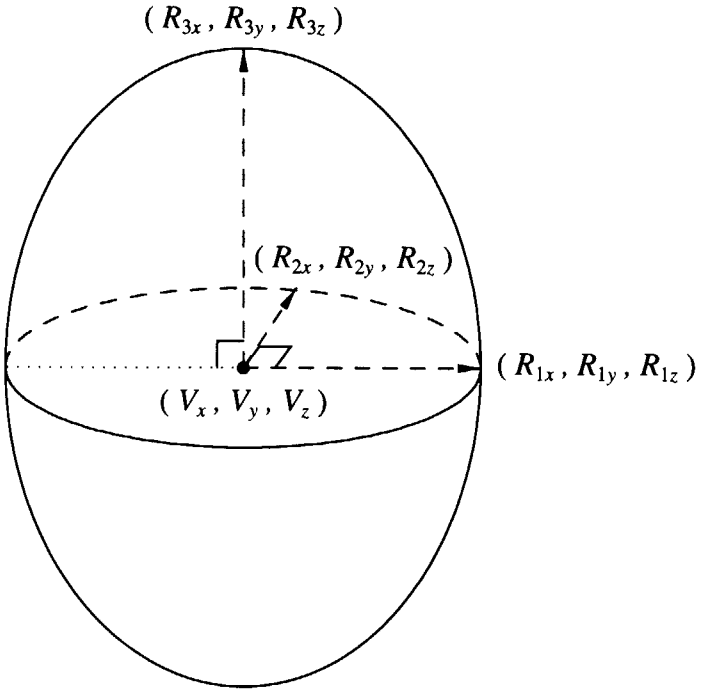

Figure 31. Illustration of a general ellipsoid (GEL).

4.1.12 Truncated Right Elliptic Cone (QUA)

QUA is defined by the two ellipses parallel to the X-Y plane. The QUA is represented in the mathematical manner as follows.

$$(ax + by + cz)^2 + (dx + ez + f)^2 = (gz + h)^2 \tag{21}$$

or

$$\left\{\frac{x + (bz + c)/a}{(gz + h)/a}\right\}^2 + \left\{\frac{y + (ez + f)/d}{(gz + h)/d}\right\}^2 = 1 \tag{22}$$

Here, the distances between the origin O and the centers of the two ellipses P and Q are given as,

$$\overrightarrow{OP} = \left(-\frac{bZ_1+h}{a}, -\frac{eZ_1+f}{d}, Z_1\right), \tag{23}$$

$$\overrightarrow{OQ} = \left(-\frac{bZ_2+h}{a}, -\frac{eZ_2+f}{d}, Z_2\right). \tag{24}$$

And the half length of the major and minor axes of the two ellipses are given as,

$$(|PA|, |PB|) = \left(\frac{gZ_1+h}{a}, -\frac{gZ_1+h}{d}\right), \tag{25}$$

$$(|QA'|, |QB'|) = \left(\frac{gZ_2+h}{a}, -\frac{gZ_2+h}{d}\right). \tag{26}$$

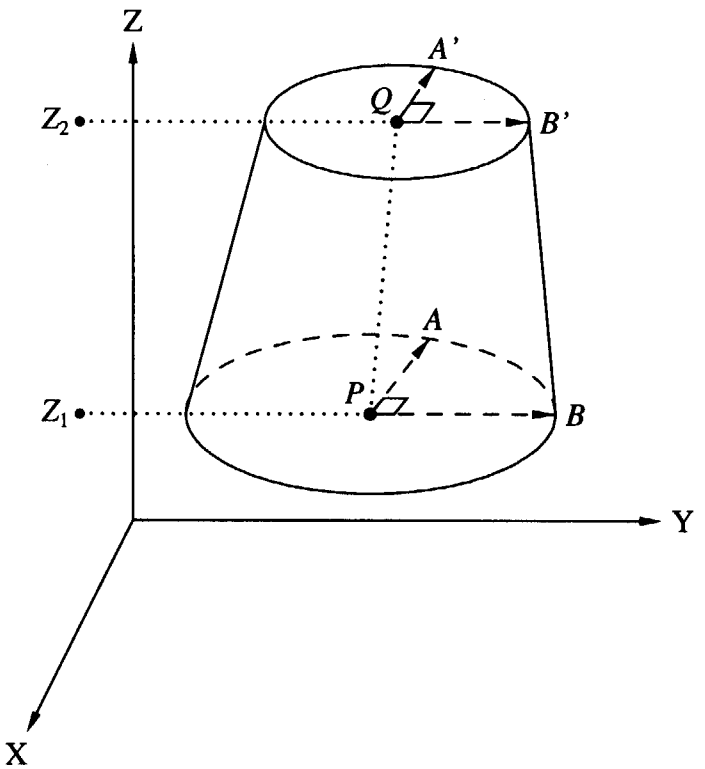

Figure 32. Illustration of a truncated right elliptic cone (QUA).

4.1.13 Torus (TOR)

TOR is defined by the rotation of a right ellipse having its center apart from the origin (X_0, Y_0, Z_0) at the distance R. The values of the major and minor axes of the ellipse a and b are given as input data with the angles for the start and end of the rotation, θ_1 and θ_2, respectively. The center axis for the rotation is specified with a parameter F_{xyz}. $F_{xyz} = 1$ means that θ_1 and θ_2 are the angles around the X-axis, in other words on the Y-Z plane, $F_{xyz} = 2$ on the X-Z plane, $F_{xyz} = 3$ on the X-Y plane, respectively.

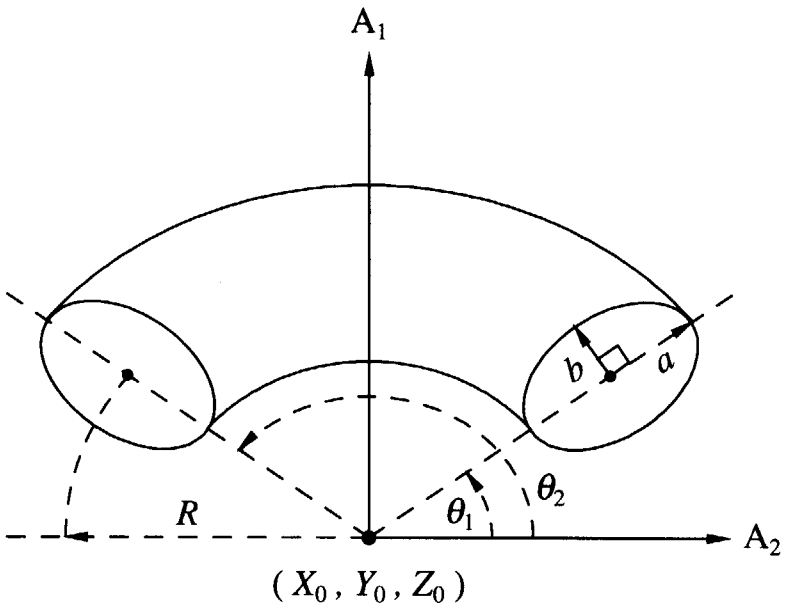

Figure 33. Illustration of a torus (TOR).

4.1.14 Plane Defined by Hesse Formula (P)

P is defined by the plane described by Hesse equation. The vector normal to the plane is given by (H_x, H_y, H_z) and the distance from the origin is D. In the region section, the "+" and "−" characters specify the half space divided by the plane. The "+" defines the half space normal side along the vector (H_x, H_y, H_z) and the "−" the opposite side.

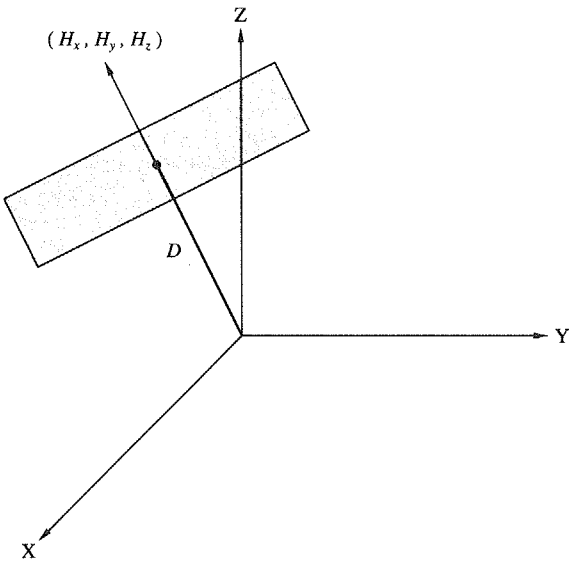

Figure 34. Illustration of a plane defined by Hesse formula (P).

4.1.15 Plane Normal to X-axis (PX)

PX is defined by the plane normal to X-axis. The distance from the origin is given by D. In the region section, the "+" and "−" characters specify the half space divided by the plane. The "+" defines the half space normal side along the vector $(1, 0, 0)$ and the "−" the opposite side.

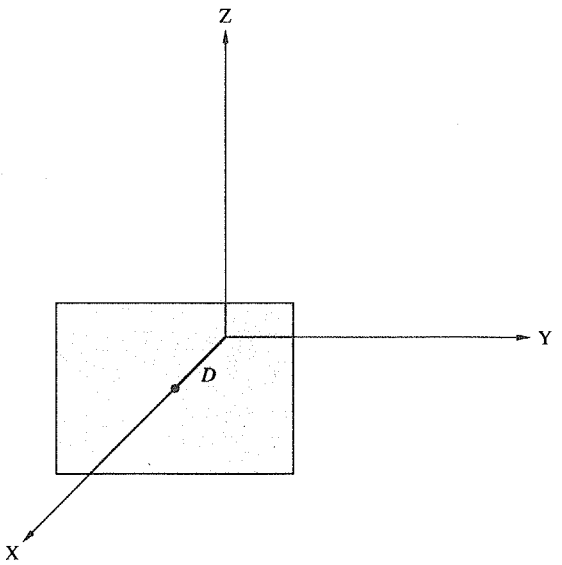

Figure 35. Illustration of a plane normal to X-axis (PX).

4.1.16 Plane Normal to Y-axis (PY)

PY is defined by the plane normal to Y-axis. The distance from the origin is given by D. In the region section, the "+" and "−" characters specify the half space divided by the plane. The "+" defines the half space normal side along the vector $(0, 1, 0)$ and the "−" the opposite side.

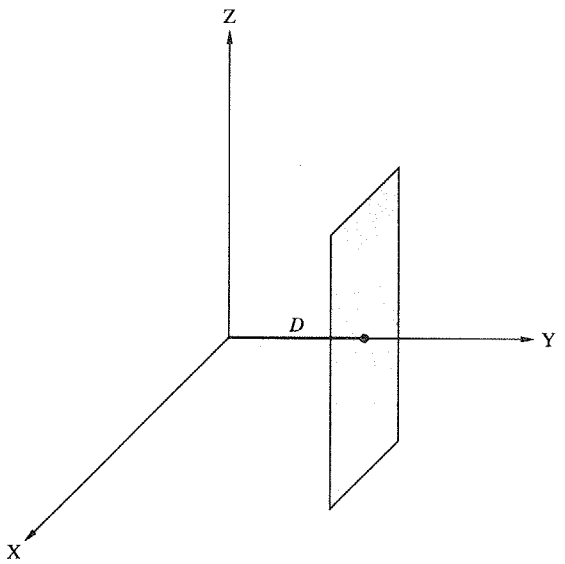

Figure 36. Illustration of a plane normal to Y-axis (PY).

4.1.17 Plane Normal to Z-axis (PZ)

PZ is defined by the plane normal to Z-axis. The distance from the origin is given by D. In the region section, the "+" and "−" characters specify the half space divided by the plane. The "+" defines the half space normal side along the vector $(0, 0, 1)$ and the "−" the opposite side.

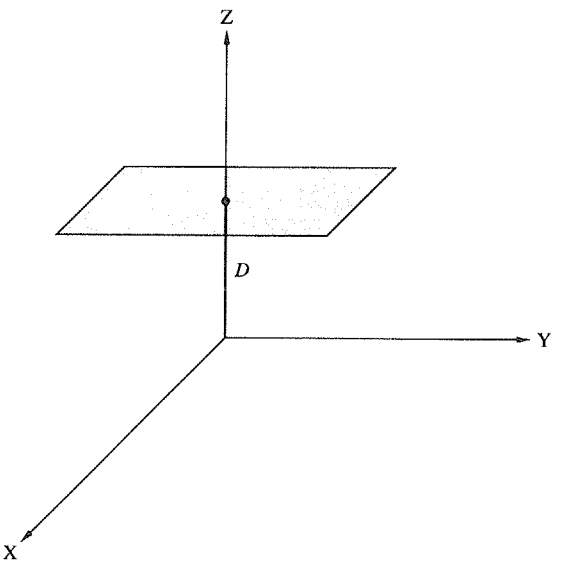

Figure 37. Illustration of a plane normal to Z-axis (PZ).

4.1.18 Plane Defined by Generalized Hesse Formula (PS)

PS is defined by the plane described by generalized Hesse equation. The vector normal to the plane is given by (H_x, H_y, H_z) and the distance from the position (V_x, V_y, V_z) is D. In the region section, the "+" and "−" characters specify the half space divided by the plane. The "+" defines the half space normal side along the vector (H_x, H_y, H_z) and the "−" the opposite side.

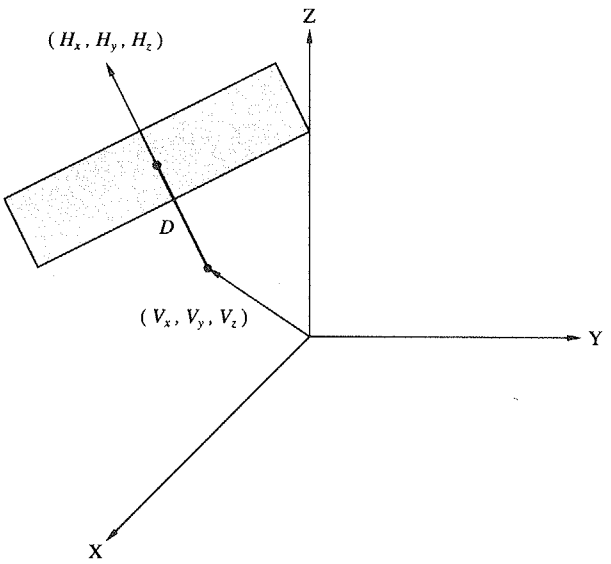

Figure 38. Illustration of a plane defined by generalized Hesse formula (PS).

4.1.19 Cylinder (C)

C is defined by the infinite right circular cylinder which center line is on the position (V_x, V_y, V_z), the direction of the cylinder is given by (H_x, H_y, H_z) and the radius is R. In the region section, the "+" and "−" characters define the inside and outside spaces of the cylinder.

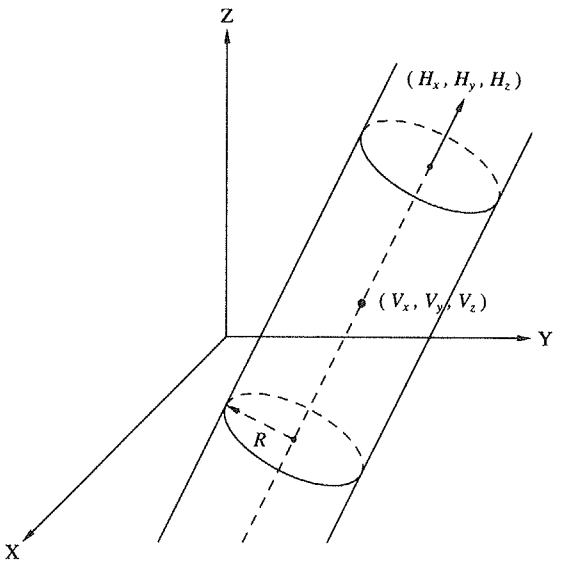

Figure 39. Illustration of an infinite cylinder (C).

4.1.20 Cylinder Parallel to X-axis (CX)

CX is defined by the infinite right circular cylinder parallel to X-axis which center line is on the position (V_x, V_y, V_z) and the radius is R. In the region section, the "+" and "−" characters define the inside and outside spaces of the cylinder.

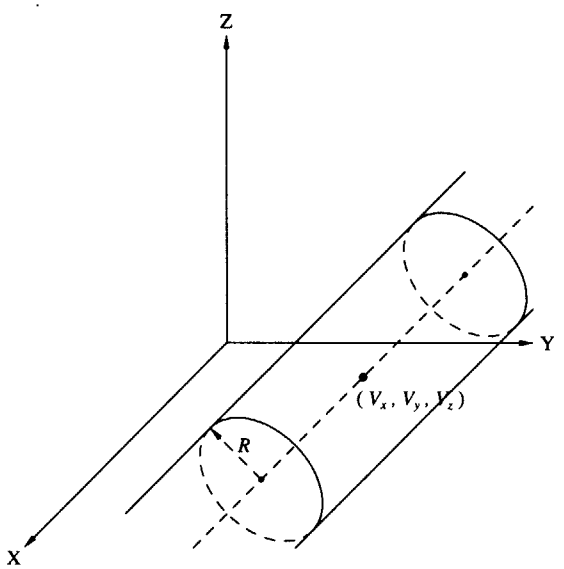

Figure 40. Illustration of an infinite cylinder parallel to X-axis (CX).

4.1.21 Cylinder Parallel to Y-axis (CY)

CY is defined by the infinite right circular cylinder parallel to Y-axis which center line is on the position (V_x, V_y, V_z) and the radius is R. In the region section, the "+" and "−" characters define the inside and outside spaces of the cylinder.

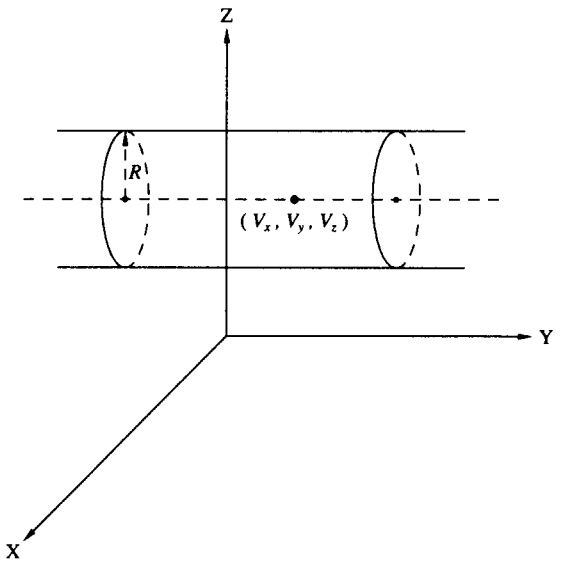

Figure 41. Illustration of an infinite cylinder parallel to Y-axis (CY).

4.1.22 Cylinder Parallel to Z-axis (CZ)

CZ is defined by the infinite right circular cylinder parallel to Z-axis which center line is on the position (V_x, V_y, V_z) and the radius is R. In the region section, the "+" and "−" characters define the inside and outside spaces of the cylinder.

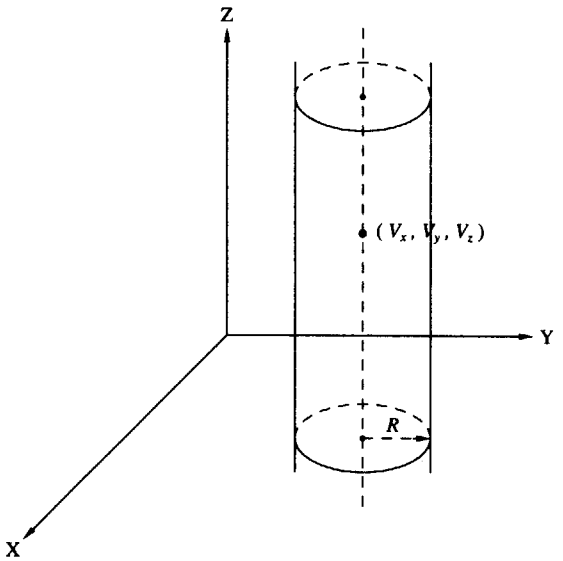

Figure 42. Illustration of an infinite cylinder parallel to Z-axis (CZ).

4.2. Region Description

Region is defined by the combination of bodies using the set operator. The blank character " " and the "OR" notation are the operators meaning the intersection (AND) and the union (OR) of the two bodies, respectively. The "+" and "−" characters are also attached ahead of the body identification number. The "+" character means that the region is inside the body. In this case, it is possible to omit the "+" character. The "−" character, on the other hand, means that the region is outside the body. There is no limitation to the number of bodies used to define a region. Some examples of region description are shown in the following Table 6 and Figure 43.

Table 6. Examples of the region description.

definition	explanation
+2 +3	The region is the intersection of "the inside of the body 2" and "the inside of the body 3"
+2 −3	The region is the intersection of "the inside of the body 2" and "the outside of the body 3"
+2 OR +3	The region is the union of "the inside of the body 2" and "the inside of the body 3"
+1 −2 −3	The region is the intersection of "the inside of the body 1" and "the outside of the body 2" and "the outside of the body 3"

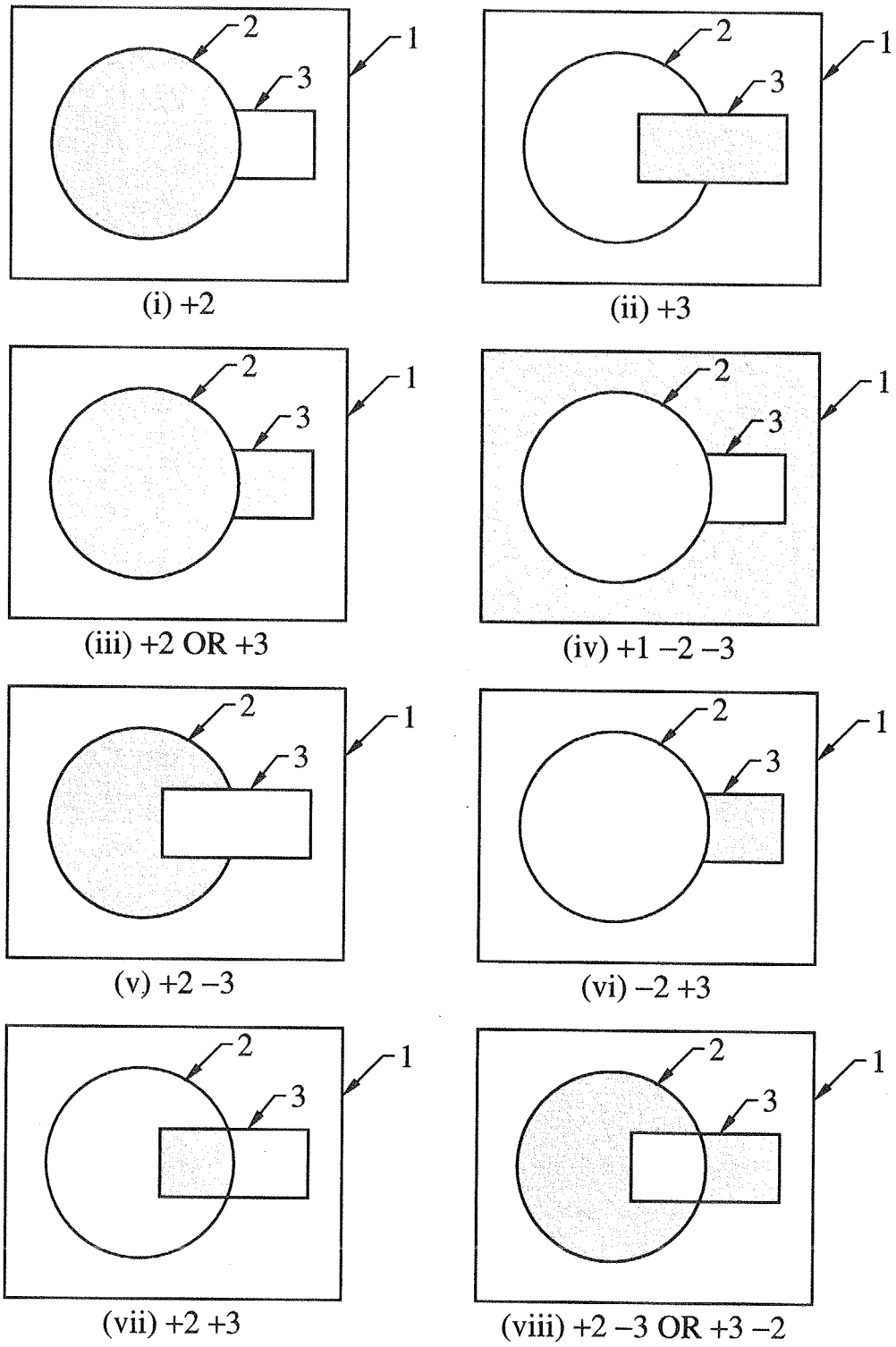

Figure 43. Examples of the region description.

4.3. Array and Universe Description

In NMTC/JAM, any geometries can be modeled using only the above ordinary CG manner. For special cases to define geometry configurations composed of assemblies of some elements such as reactor core composed of fuel assemblies, human body phantom and so on, the features of MARS, "array" and "universe", are useful to describe such geometries. The relation between arrays and universes is shown in Figure 44.

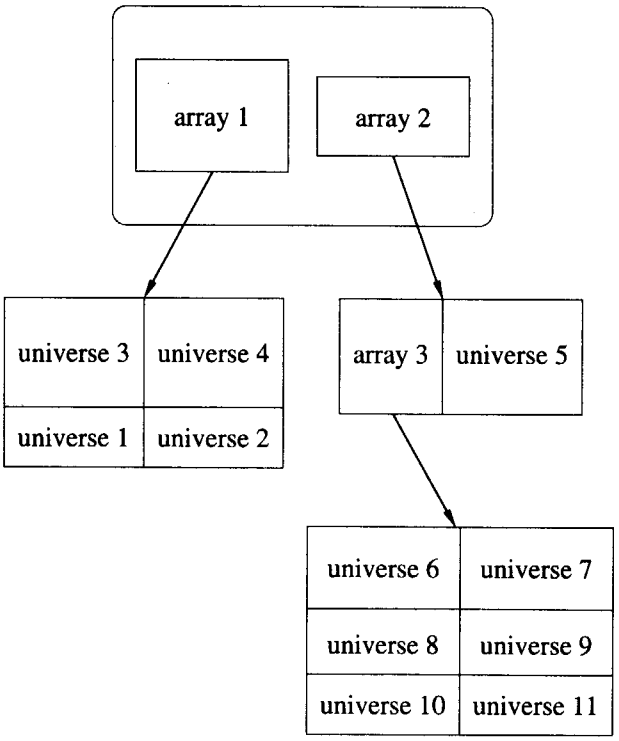

Figure 44. Relation between arrays and universes described by the geometry model MARS.

The array is a regular rectangular lattice composed of rectangular cells of arbitrary content. The size of the array is defined by giving the number of cells along X, Y and Z axes, respectively. The arrays are sequentially labelled from 1 as they are entered in the input card. In order to terminate the entries for arrays, zero should be supplied in the card. After this termination input, a single integer parameter is required to define the input specification method, in other words, the data format.

An example of the input card for an array is shown as follows.

List 4.1 ● Example of the array input

```
1:   10   10   1   7   4   2   6   6   1   0   0
```

This example describes three arrays. Array 1 will be a 10 by 10 by 1 array. Array 2 will be a 7 by 4 by 2 array. And, the array 3 will be a 6 by 6 by 1. The next zero is for terminating the

array size input. The last zero indicates that the free-form input specification method is selected to define the array contents. The options of the input specification methods are explained in Section 9. The user must give the contents of each cell of each array according to the data format selected by the input specification option.

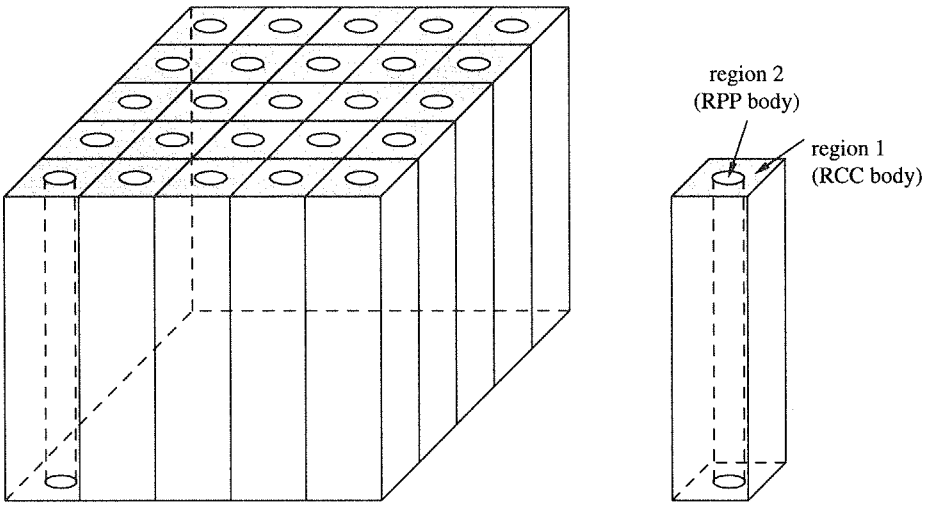

Figure 45. Array (left-hand side) and Universe (right-hand side).

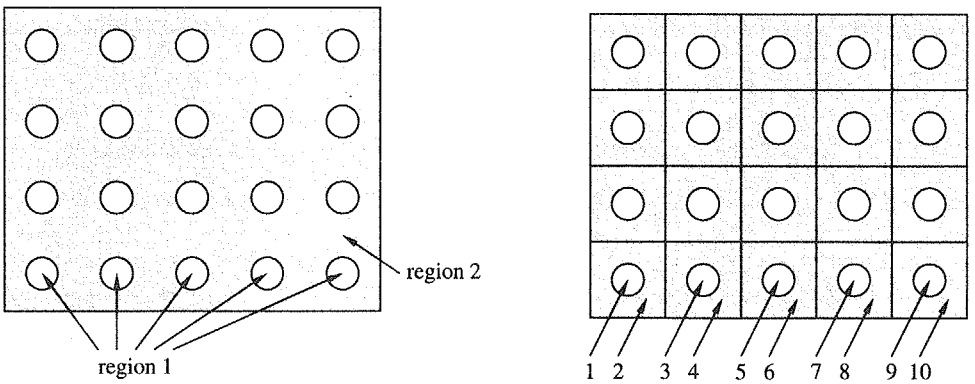

Figure 46. Left-hand side is the region number assignment in case that an array is composed of single universe. Right-hand side is the region number assignment in case that an array is composed of different universes.

The universe is defined as the separate unique geometry modeled by CG and so can describe nested structures and array system. A universe has a separate independent dimension in space defined by a collection of input regions bounded by a special boundary media. Every input region resides in the universe and must include the universe number containing the region. The contents of the universe are arbitrary, and the universe may contain an array in one or more

of its input region. The most global universe is called as absolute universe and is designated as universe 0. All universes with exception of the absolute universe must be surrounded with the outer void. The coordinates in the universe can be defined independent of those in other universes. The universe number is given with a positive integer.

Figure 45 shows a sample geometry configuration which can be described using the function of array and universe. As seen in the Figures 46, the array can be defined in different ways using a single universe or different universes. The geometry description data for each case are given in the following.

List 4.2 ● Example of input data for an array structure composed of single universe

```
 1: [ B o d y ]
 2:  idbg = 0 ;  ibod = 1 ;  naz =  0
 3:   num   sym     def
 4:     1   rcc    0.0    0.0    0.0    0.0    0.0    10.0    1.0
 5:     2   rpp   -2.0    2.0   -2.0    2.0    0.0    10.0
 6:     3   rpp    0.0   20.0    0.0   16.0    0.0    10.0
 7:     4   sph    0.0    0.0    0.0 1000.0
 8: [ R e g i o n ]
 9:   num    mat   uni    sym   def
10:     1      1     1    det   +1
11:     2      2     1    col   +2 -1
12:     3  -1000     1    out   -2
13:     4     -1     0    all   +3
14:     5      0     0    vac   +4 -3
15: [ A r r a y ]
16:     5      4     1     0     2
17:     101$$   20r1   t
18:     0
```

List 4.2 shows the input data for an array structure composed of single universe, which corresponds to the left-hand side of Figure 46. It is possible to describe the geometry with very simple input data by setting single universe as shown in the right-hand side of Figure 46 and List 4.2. However, the information about each cell (element) in the array can not be obtained by the NMTC/JAM calculation because the region number is not assigned to individual cell but to region of the universe. The present NMTC/JAM recognizes the region number of an universe as that for tally function. If the user wants to know the information about each cell of the array, the cell must be composed of a universe different from the others as shown in the right-hand side of Figure 46 and List 4.3.

List 4.3 ● Example of input data for an array structure composed of different universe

```
 1:  [ B o d y ]
 2:   idbg = 0 ;  ibod = 1 ;  naz =  0
 3:    num   sym     def
 4:      1   rcc    0.0    0.0    0.0    0.0    0.0    10.0    1.0
 5:      2   rpp   -2.0    2.0   -2.0    2.0    0.0    10.0
 6:      3   rpp    0.0   20.0    0.0   16.0    0.0    10.0
 7:      4   sph    0.0    0.0    0.0 1000.0
 8:  [ R e g i o n ]
 9:    num    mat   uni    sym   def
10:      1      1     1    d01   +1
11:      2      2     1    c01   +2 -1
12:      0  -1000     1    o01   -2
13:      3      1     2    d02   +1
14:      4      2     2    c02   +2 -1
15:      0  -1000     2    o02   -2
16:      5      1     3    d03   +1
17:      6      2     3    c03   +2 -1
18:      0  -1000     3    o03   -2
19:      7      1     4    d04   +1
20:      8      2     4    c04   +2 -1
21:      0  -1000     4    o04   -2
22:      9      1     5    d05   +1
23:     10      2     5    c05   +2 -1
24:      0  -1000     5    o05   -2
25:     11      1     6    d06   +1
26:     12      2     6    c06   +2 -1
27:      0  -1000     6    o06   -2
28:     13      1     7    d07   +1
29:     14      2     7    c07   +2 -1
30:      0  -1000     7    o07   -2
31:     15      1     8    d08   +1
32:     16      2     8    c08   +2 -1
33:      0  -1000     8    o08   -2
34:     17      1     9    d09   +1
35:     18      2     9    c09   +2 -1
36:      0  -1000     9    o09   -2
37:     19      1    10    d10   +1
38:     20      2    10    c10   +2 -1
39:      0  -1000    10    o10   -2
40:     21      1    11    d11   +1
41:     22      2    11    c11   +2 -1
42:      0  -1000    11    o11   -2
43:     23      1    12    d12   +1
44:     24      2    12    c12   +2 -1
45:      0  -1000    12    o12   -2
46:     25      1    13    d13   +1
47:     26      2    13    c13   +2 -1
48:      0  -1000    13    o13   -2
49:     27      1    14    d14   +1
50:     28      2    14    c14   +2 -1
51:      0  -1000    14    o14   -2
52:     29      1    15    d15   +1
53:     30      2    15    c15   +2 -1
54:      0  -1000    15    o15   -2
55:     31      1    16    d16   +1
56:     32      2    16    c16   +2 -1
57:      0  -1000    16    o16   -2
58:     33      1    17    d17   +1
59:     34      2    17    c17   +2 -1
60:      0  -1000    17    o17   -2
```

```
61:     35      1    18    d18   +1
62:     36      2    18    c18   +2 -1
63:      0  -1000    18    o18   -2
64:     37      1    19    d19   +1
65:     38      2    19    c19   +2 -1
66:      0  -1000    19    o19   -2
67:     39      1    20    d20   +1
68:     40      2    20    c20   +2 -1
69:      0  -1000    20    o20   -2
70:      0     -1     0    arr   +3
71:      0      0     0    vac   +4 -3
72:  [ A r r a y ]
73:      5      4     1     0     2
74:      101$$   18i1 20  t
75:      20*0
```

The input data of 18 line of List 4.2 and 75 line of List 4.3 are the parameters which define the type of universe. One can define the "simple" universe by 1 and "combinatorial universe" by 0. A "simple" universe is a universe composed of concentric regions, where every region completely surrounds the region inside of it. Furthermore, input regions in a simple universe may be only one code region and may be described by only one or two bodies. Figures 47 and 48 show the sketches of a simple and a combinatorial universe.

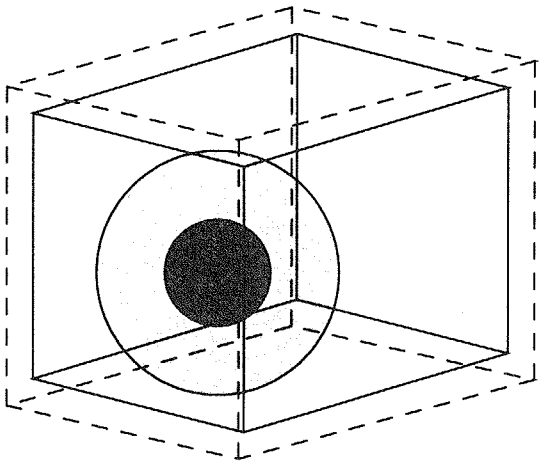

Figure 47. Illustration of a simple universe.

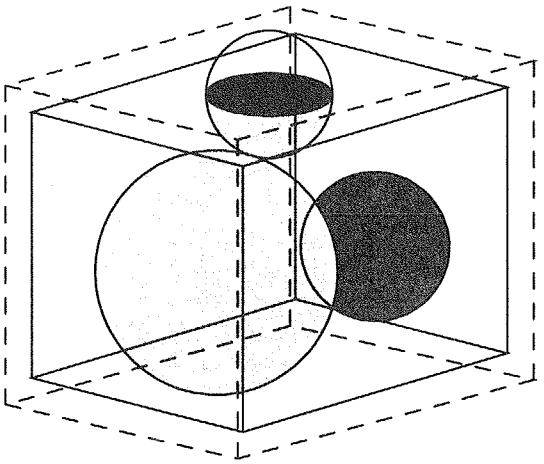

Figure 48. Illustration of a combinatorial universe.

5. Importance Sampling

NMTC/JAM implements the function that can calculate effectively the spectrum of particles with an importance sampling technique by setting a weight to a particle in each region. The statistical errors are also calculated taking account of the weight of a particle. The present importance sampling is carried out in the following procedure when a particle having a weight of "w_t" moves from the current region with an importance of "w_1" to the next region with that of "w_2".

(1) The ratio of the importance of neighboring regions is calculated as $r = w_2/w_1$.

(2) If $r > 1.0$ then, a sampling with a uniform random number x is performed. If $x \geq r - n$ where n is the integer with the condition of $n \leq r \leq n + 1$, a parent particle is divided into n particles with a weight of w_t/r, and then particles with the number of $n - 1$ are banked for the next transport calculation. If $x < r - n$, the parent particle is divided into $n + 1$ particles with the weight of w_t/r, and the particles with the number of n are banked. Here, the information of the particles such as energy, position and directional cosines are succeeded with the one of parent particles except for the weight information. This procedure is generally called as splitting technique.

(3) If $r < 1.0$ then, a sampling is performed on the basis of a uniform random number. If the selected random number is less than r, the particle is survived and simulated with a weight of w_t/r. If the random number is greater than r, the transport calculation of the particle is terminated. This procedure is generally called as Russian roulette technique.

(4) If $r = 1.0$ ($w_2 = w_1$) or the importance of the next region is zero, the particle goes to the next region keeping the current weight w_t .

If the importance sampling is executed in the calculation, the summary table of the importance sampling is printed out.

6. Tally Function

In NMTC/JAM, the following six tallies are prepared for statistics calculations:

(1) track length tally
(2) surface crossing tally
(3) nuclide yield tally
(4) heat tally
(5) time tally
(6) star density tally

In these tally functions, the user can use three kinds of geometry mesh, the region mesh of CG, r-z scoring mesh and xyz scoring mesh. The detail of input parameters and output format for the tallies is explained in Section 9. In this section, we describe the tally functions which are booked in the calculations.

6.1. Track Length Tally

The track length tally calculates the flux in a certain geometry mesh by the following representation:

$$\phi_j(E) = \frac{\sum_i w_i(E) \cdot l_i}{V_j \cdot dE \cdot N}, \tag{27}$$

where

$w_i(E)$: weight of i-th particle with energy E
l_i : track length of i-th particle in the specified mesh j [cm]
V_j : volume of the geometry mesh j [cm^3]
dE : width of energy bin [MeV or Lethargy]
N : number of source particles

6.2. Surface Crossing Tally

The surface crossing tally calculates the current or flux by the surface crossing of a certain geometry mesh by the following representation:

$$J_k(E) = \frac{\sum_i w_i(\mu, E)}{S_k \cdot dE \cdot N} \quad : \text{current}, \tag{28}$$

$$\phi_k(E) = \frac{\sum_i w_i(\mu, E)}{S_k \cdot dE \cdot N} \cdot \frac{1}{|\mu|} \quad : \text{flux}, \tag{29}$$

where

$w_i(\mu, E)$: weight of i-th particle crossing the specified surface k with directional cosine μ and energy E

S_k : area of the surface k [cm^2]

dE : width of energy bin [MeV or Lethargy]

N : number of source particles

6.3. Nuclide Yield Tally

The nuclide yield tally calculates the nuclide production yield in a certain geometry mesh by the following representation:

$$Y_k(l,m) = \frac{\sum_i w_i \cdot n(l,m)}{V_k \cdot N}, \tag{30}$$

where

w_i : weight of i-th particle

$n(l,m)$: number of nucleus l production from the mother nucleus m

V_k : volume of the geometry mesh k [cm^3]

N : number of source particles

6.4. Heat Tally

The heat tally calculates the deposit energy by the recoil, ionization, and the other processes in a certain geometry mesh by the following representation:

$$E_r(k) = \frac{\sum_i w_i \cdot e_r(i)}{V_k \cdot N} \quad : \text{recoil}, \tag{31}$$

$$E_l(k) = \frac{\sum_i w_i \cdot e_l(i)}{V_k \cdot N} \quad : \text{ionization}, \tag{32}$$

where

w_i : weight of i-th particle

$e_r(i)$: recoil energy of nuclei induced by i-th particle

$e_l(i)$: ionization energy loss of i-th particle

V_k : volume of the geometry mesh k [cm^3]

N : number of source particles

As for the other processes, we consider the energy of stopped particles and remaining excitation energy of residual nuclei.

6.5. Star Density Tally

The star density tally calculates the density of the nuclear reactions in a certain geometry mesh by the following representation:

$$S_j = \frac{\sum_i w_i(E) \cdot n_i}{V_j \cdot dE \cdot N}, \tag{33}$$

where

$w_i(E)$: weight of i-th particle with energy E

n_i : number of nuclear reactions induced by i-th particle

V_j : volume of the geometry mesh j [cm^3]

dE : width of energy bin [MeV or Lethargy]

N : number of source particles

6.6. Time Tally

The time density tally calculates the time when the particle reaches the outer region or the cut off energy in a certain geometry mesh by the following representation:

$$T_j = \frac{\sum_i w_i(E) \cdot n_i}{V_j \cdot dE \cdot N}, \tag{34}$$

where

$w_i(E)$: weight of i-th particle with energy E

n_i : number of escape or cut off particles

V_j : volume of the geometry mesh j [cm^3]

dE : width of energy bin [MeV or Lethargy]

N : number of source particles

6.7. Statistical Error of Tally

In the statistical calculation of tallies, we estimate the statistical error σ by

$$\sigma = \sqrt{\frac{\sum_i X_i \cdot W_i^2}{(\sum_i X_i \cdot W_i)^2} - \frac{1}{N}} \tag{35}$$

where

W_i : weight of i-th event

X_i : scoring quantity

N : number of source particles

7. Particle Transport in Magnetic Field

The equation of motion for charged particle in an external magnetic field M is described by,

$$\begin{pmatrix} x \\ x' \end{pmatrix} = M \begin{pmatrix} x_0 \\ x'_0 \end{pmatrix}, \tag{36}$$

where x, x' are the coordinate [mm] and direction [mrad], and x_0, x'_0 represent those before the transport, respectively. We consider the dipole and quadrupole magnet for the magnetic field M. The M is given by the each element as,

$$M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}. \tag{37}$$

The detail of the procedure of particle transport in magnetic field is discussed in Sec.2.3.

7.1. Dipole case

In the case of dipole magnet and bending in x-direction, the beam is only drifted in y-direction. Thus the matrix elements of M for y-direction is given by

$$M_{11} = M_{22} = 1, \tag{38}$$

$$M_{12} = L, \tag{39}$$

$$M_{21} = 0, \tag{40}$$

where L [m] is the length of the magnet. On the other hand, those for x-direction are

$$M_{11} = \cos(kL), \tag{41}$$

$$M_{12} = \frac{1}{k}\sin(kL), \tag{42}$$

$$M_{21} = -k\sin(kL), \tag{43}$$

$$M_{22} = \cos(kL), \tag{44}$$

$$B\rho = 33.356p, \tag{45}$$

$$k = \frac{1}{\rho}, \tag{46}$$

where B [kG] represents the strength of magnetic field, p [GeV/c] is the momentum of the particle.

7.2. Quadrupole case

In the case of quadrupole magnet and the magnetic field B is positive, the beam is focused in x-direction. The matrix elements of M for x-direction is given by

$$M_{11} = \cos(kL), \tag{47}$$

$$M_{12} = \frac{1}{k}\sin(kL), \tag{48}$$

$$M_{21} = -k\sin(kL), \tag{49}$$

$$M_{22} = \cos(kL), \tag{50}$$

and for y-direction,

$$M_{11} = \cosh(kL), \tag{51}$$
$$M_{12} = \frac{1}{k}\sinh(kL), \tag{52}$$
$$M_{21} = k\sinh(kL), \tag{53}$$
$$M_{22} = \cosh(kL), \tag{54}$$

where

$$k^2 = \frac{B}{a}\frac{1}{33.3556}\cdot p, \tag{55}$$

with the strength of the magnetic field B [kG], and the half gap a [m] of the magnet. For the case of the negative B, the above expressions are exchanged in x and y direction.

8. Nuclear Reaction Calculation

By making use of the components included in NMTC/JAM, we have prepared the calculation routines of the elementary processes. The elementary processes which can be calculated in the code are the followings.

(1) double differential cross sections of nuclear reactions
(2) total, elastic and non-elastic cross sections of nuclear reactions
(3) angular distribution of elastic collisions
(4) particle-particle cross sections

The detail of input parameters and output format for the nuclear reaction calculation is explained in Section 9.

9. User Guide to NMTC/JAM

9.1. Installation

The program of NMTC/JAM is designed for the computers of Pentium, Dec alpha, Hewlett Packard UX-A9000, SPARC stations and other computers with a FORTRAN77 compiler. The NMTC/JAM code has been tested in DEC workstation, SUN workstation, Hewltt Packard workstation, and Pentium personal computer with WINDOWS and Linux OS.

9.1.1 Source, include and data files

In order to compile the NMTC/JAM code, the following source files and include files are necessary.

List 9.1 ● Source files

anal-002.f	analyz.f	bert-bl0.f	bert-bl1.f	bert-bl2.f	bert.f
bertin.f	celimp.f	dataup.f	dklos.f	energy.f	erup.f
erupin.f	fissn.f	gamlib.f	gem.f	gemset.f	getflt.f
gomprp.f	gomsor.f	isobert.f	isodat.f	jam.f	jamana.f
jambuu.f	jamcoll.f	jamcross.f	jamdat.f	jamdec.f	jamhard.f
jamhij.f	jamin.f	jampdf.f	jamsoft.f	jbook.f	magtrs.f
main.f	mars01.f	mars02.f	mars03.f	mars04.f	marslib.f
masdis.f	mdp-uni.f	mdp-win.f	ncasc.f	nelst.f	nevap.f
nreac.f	ovly12.f	ovly13.f	partrs.f	pyjet.f	pysigh.f
pythia.f	randmc.f	range.f	read00.f	read01.f	read02.f
sdml.f	sors.f	stat.f	sublib.f	talls01.f	talls02.f
talls03.f	talls04.f	unix.f	update.f	usrsors.f	utl01.f
utl02.f	utl03.f	utl04.f	utlnmtc.f	wrnt12.f	wrnt13.f

In the above source files, the machine dependent source is only **mdp-uni.f, mdp-win.f**. You should choose **mdp-win.f** for WINDOWS machine, **mdp-uni.f** for the other machines. This subroutine gets the date, time and CPU time of the execution.

The include files, which mainly describe the dimensions of the common block, is the followings.

List 9.2 ● Include files

bert.inc	cm_memar.inc	gamlib.inc	jam1.inc	jam2.inc
jam3.inc	param.inc	param00.inc	param01.inc	param02.inc

In run time, the data file for the gamma emission from the residual nuclei is necessary. Its name is **trxcrd.dat**. You can locate this file in any directory which is specified by the input parameter.

9.1.2 Makefile

An example of the makefile to compile the source is given below.

List 9.3 ● Makefile

```
#-----------------------------------------------------------------------
#       @(#)Makefile  2000/10/23
#       Makefile for NMTCJAM100
#-----------------------------------------------------------------------
##### sors #####
#-----------------------------------------------------------------------
SRC = \
analyz.f   bert.f     \
nreac.f    celimp.f   dataup.f   dklos.f    erupin.f   \
erup.f     fissn.f    getflt.f   gomprp.f   \
gomsor.f   isobert.f  isodat.f   main.f     nevap.f    \
read01.f   ovly12.f   \
ncasc.f    randmc.f   range.f    energy.f   ovly13.f   \
sors.f     masdis.f   \
partrs.f   sublib.f   update.f   wrnt12.f   wrnt13.f   \
utlnmtc.f  gamlib.f   bertin.f   \
nelst.f    bert-bl0.f bert-bl1.f bert-bl2.f \
jamin.f    jam.f      jamdat.f   jamcoll.f  jamdec.f   \
jamcross.f jampdf.f   jamsoft.f  jamhij.f   jamhard.f  \
jambuu.f   jamana.f   unix.f     pyjet.f    pythia.f   \
pysigh.f   jbook.f    sdml.f     utl01.f    utl02.f    \
read00.f   read02.f   utl03.f    utl04.f    \
gem.f      gemset.f   talls01.f  talls02.f  usrsors.f  \
magtrs.f   talls03.f  talls04.f  mdp-uni.f  \
mars01.f   mars02.f   mars03.f   mars04.f   \
anal-002.f
#-----------------------------------------------------------------------
##### set executable file name #####
#-----------------------------------------------------------------------
 EXE = ../nmtcjam
#-----------------------------------------------------------------------
##### set compile options #####
#-----------------------------------------------------------------------
#Alpha
  FFLAGS = -g -O4 -align dcommons -convert big_endian -warn noalignments -warn nousage
  FC = f77
#-----------------------------------------------------------------------
#Linux pgf
# FFLAGS = -O0
# FC = /usr/local/pgi/linux86/bin/pgf77
#-----------------------------------------------------------------------
#Linux g77
# FFLAGS = -O0 -fno-automatic
# FC = /usr/bin/g77
#-----------------------------------------------------------------------
#Sun
# FFLAGS = -O
# FC = f77
#-----------------------------------------------------------------------
#HP
# FFLAGS = +DA1.1 +O0 +e +U77 +Onolimit
# FC = f77
#-----------------------------------------------------------------------
LFLAGS = $(FFLAGS)
CFLAGS = -O
CC = cc
##### make executables #####
$(EXE): $(SRC:.f=.o)
:  $(FC) -o $(EXE) $(LFLAGS) $(SRC:.f=.o)
##### make objects #####
```

```
61:  .f.o:
62:  :  $(FC) -c $(FFLAGS) $<
63:  .c.o:
64:  :  $(CC) -c $(CFLAGS) $<
65:  ##### clean-up #####
66:  clean:
67:  :  rm -f core $(EXE:=.trace) *~ "#"*
68:  objclean: clean
69:  :  rm -f $(SRC:.f=.o)
70:  distclean: objclean
71:  :  rm -f $(EXE)
```

9.1.3 Execution shell

The special execution shell is not necessary. The NMTC/JAM program reads the input parameters from the unit 5 (standard input unit) and writes the summary and error information on unit 6 (standard output unit). The other files, which are necessary for the input data and output data, are specified in the input parameters.

Therefore, the most simple command to execute NMTC/JAM is

List 9.4 ● command line to execute NMTC/JAM

```
nmtcjam < input.filename > output.filename
```

where `nmtcjam` is the executable file of NMTC/JAM.

However, this command line is not available in WINDOWS, since the rewind command cannot be used in the standard input file in WINDOWS. Thus we have prepared the other method to specify the input file. If the first line of the standard input is written as

List 9.5 ● the first line of the standard input

```
file = input.file
```

In this case, the program does not read the standard input any more, but it opens the `input.file` and reads the input parameters from this `input.file`.

You can use both methods, but should use the latter one for WINDOWS machine.

9.2. Format of Input File

Input data of NMTC/JAM consist of the "section" starting from `[******]`. The name of section is listed in Table 7. The order of the sections is free.

9.2.1 List of Sections

There are 17 sections, listed in the next table, in which the program reads the input parameters. Therefore, the data from the top line to the first section, nor after the end section to the end of file is neglected by the program.

Table 7. List of sections.

name	explanation
[title]	defines the title of the calculation
[parameters]	defines the parameters which control the transport calculation
[source]	define the information of the source particles
[material]	defines the material of the system
[body]	defines the body of the system
[region]	defines the region of the system
[allay]	defines the allay of the geometry configuration
[importance]	defines the importance
[volume]	defines the volume of the region
[magnetic field]	defines the magnetic field of the region
[t-track]	defines the parameters of track length tally
[t-cross]	defines the parameters of surface crossing tally
[t-yield]	defines the parameters of nuclide yield tally
[t-heat]	defines the parameters of heat tally
[t-time]	defines the parameters of time tally
[t-star]	defines the parameters of star density tally
[end]	indicates the end of input file

9.2.2 Rules of Reading the Input Data

(1) Characters of upper and lower case and blank

We do not distinguish the upper and lower case except for the file name. The blank characters at the top and the bottom of line are neglected. In section name, the blank is neglected.

(2) Sequential lines

If the character "\" is put at the bottom of line, the sequential line is connected to the previous one. In the `def` of `[body]` section and `def` of `[region]` section, the sequential line is automatically recognized, then "\" is not necessary.

(3) Line connection

Some lines are connected into one line by the character ";". For a example,

```
idbg = 0 ;  ibod = 1 ;  naz =  0
```

However, for the case of fixed format appeared in mesh description, this line connection is not available.

(4) Comment characters

The characters "#", "%", "!", and "$" are defined as comment characters. After these characters up to the bottom of line, all characters are neglected as a comment.

(5) Blank lines

Blank line is skipped in reading data. The line with the comment characters at its top is

also skipped.

(6) Skip a section

If "off" is put after the section name like "[******] off", this section is skipped until the other section is appeared.

(7) Skip a part of section

If "qp:" is put on the top of line in a middle of the section, the remaining part of the section is skipped until the other section appears.

(8) Skip the remaining part

If "q:" is put on the top of line in any place, the remaining part of the input data is skipped. This is equivalent to "[end]" section.

9.2.3 Include Files

At any place of the input file, one can include the other file or a part of the other file. The format of the include file is given as,

infl: { *file.name* } [$n_1 - n_2$]

where the inside of { } is the file name and [ ] specifies the start line and the end line of the include file. The line number can be omitted. For that case, all of the file is included. For the line specification, the following expressions are allowed,

[n_1-]
[$-n_2$]

For the upper case, from n_1 line to the end of file, for the lower case, from the top of the file to n_2 line, are included.

The procedure of including file can be nested any times. After reading the include file, the program returns to the previous place of the parent file.

9.2.4 Constants Defined by the User

The constants defined by the user can be used in any place of the input data. The format of the definition of the constant is given by,

set: c1[*52.3*] c2[*2* * pi] c3[c1 * *1.e-8*]

The user can use the constant name from c1 to c99. The reset of the constant is possible any times, and the constant keeps its value until the constant is reset. For the case that the definition of the constant uses the other constant, (third example of above expression), the value at that moment (52.3) is set into the other constant (c3) and keep its value independently of the reset of the reference constant (c1). pi(=3.1415926535897932) is a constant prepared in the code as a default.

9.2.5 Usage of Equations

The equation can be used for numeric input. The format of the equation is FORTRAN format. The available intrinsic functions are listed in Table 8.

Table 8. Intrinsic Function.

Intrinsic Function							
FLOAT	INT	ABS	EXP	LOG	LOG10	MAX	MIN
MOD	NINT	SIGN	SQRT	ACOS	ASIN	ATAN	ATAN2
COS	COSH	SIN	SINH	TAN	TANH		

For a example,

```
param = c1 * 3.5 * sin( 55 * pi / 180 )
```

The right hand side of above definition of param is expected to be one numeric number. In this case, the blank in the equation is allowed. However, in a line where some numbers separated by blank are expected, the blank can not be used in the equation. In such case, the blank should be omitted in the equation or use the bracket like [c1 * 2 / pi].

9.3. [T i t l e] Section

In this section, the title of the calculation is defined. The title is shown on the top of the standard output file. The format of the title section is

```
[ T i t l e ]
This is a test calculation of NMTC/JAM.
Any number of title lines are allowed.
.....
```

The blank line is skipped in the title section.

9.4. [P a r a m e t e r s] Section

In this section, the parameters which control the main function of NMTC/JAM are defined. The format of the parameter description is

```
[ P a r a m e t e r s ]
para1 = number | file.name
para2 = number | file.name
.....
```

The ordering of the input parameters is free, and if the description of a certain parameter is neglected, the default value is set for the parameter. In the followings, we show the list of the parameters and their values with the meanings, where (D=***) denotes the default value.

Table 9. Main parameters (1).

parameter	value	explanation
icntl	(D=0)	main mode selection
	= 0	normal NMTC/JAM calculation
	= 1	nuclear reaction calculation
	= 2	write input file for CGVIEW
	= 3	write only input echo
	= 4	write input file for MARS-PF
	= 5	no reaction, no ionization
irskip	(D=0)	control of random number
	irskip>0	skip irskip events before starting the calculation. (for debug)
	irskip<0	skip irskip random number before starting the calculation. (for manual parallel usage)
rseed	(D=0.0)	option for initial random number
	rseed<0	set initial random number from the starting time
	rseed=0	6647299061401. is the initial random number
	rseed>0	rseed is the initial random number
maxcas	(D=10)	number of events in one batch
maxbch	(D=10)	number of batches
emin(1)	(D=1.0)	cutoff energy for proton (MeV)
emin(2)	(D=1.0)	cutoff energy for neutron (MeV)
emin(14)	(D=1.0e+10)	cutoff energy for γ (MeV)
emin(i)	(D=0.1)	cutoff energy for ityp=i particle (MeV)
	i = 3-11	ityp is listed in Table 1
ielas	(D=2)	option for elastic collision
	= 0	without any elastic collision
	= 1	without proton elastic collision
	= 2	with elastic collision
ielms	(D=100)	number of angle meshes for elastic collision
inmed	(D=1)	option for nucleon-nucleon cross section for Bertini model
	= 0	free (nmtclb25.dat)
	= 1	Cugnon old (nmtclb95.dat)
	= 2	Cugnon new (nmtclb30.dat)

Table 10. Main parameters (2).

parameter	value	explanation
nevap	(D=3)	option for evaporation model
	= 0	without evaporation model
	= 1	with DRES model
	= 2	with SDM model
	= 3	with GEM model
igamma	(D=0)	option for γ emission from residual nucleus
	= 0	without γ emission
	= 1	with γ emission
isobar	(D=0)	option for ISOBAR model
	= 0	without ISOBAR model
	= 1	with ISOBAR model
ipreeq	(D=0)	option for preequilibrium model when nevap=1
	= 0	without preequilibrium model
	= 1	with preequilibrium model
level	(D=3)	option for level density when nevap=1
	= 1	8/A
	= 2	Baba's parameter
	= 3	Igunatyuk's parameter
npidk	(D=0)	option for cutoff π^-
	= 0	force the absorption reaction
	= 1	force the decay to $\mu^- + \nu_\mu$
nspred	(D=0)	option for Coulomb spread of proton beam
	= 0	without Coulomb spread
	= 1	with original formula
	= 2	with Moliere first order formula
	= 3	with Moliere third order formula
imagnf	(D=0)	option for magnetic field
	= 0	without magnetic field
	= 1	with magnetic field
andit	(D=0)	option for angular distribution of Δ in Bertini model
	= 0	50% isotropic and 50% forward
	= 1	isotropic angular distribution
	= 2	all forward
ejamnu	(D=3500.)	threshold energy from Bertini model to JAM model for nucleon (MeV)
ejampi	(D=2500.)	threshold energy from Bertini model to JAM model for pions (MeV)
eisobar	(D=0.0)	maximum energy for ISOBAR model when isobar=1 (MeV)

Table 11. Main parameters (3).

parameter	value	explanation
incut	(D=1)	option for output of cutoff neutron
	= 0	without output
	= 1	print on file(12)
	= 2	print on file(12) with time information
igcut	(D=1)	option for output of cutoff γ
	= 0	without output
	= 1	print on file(13)
	= 2	print on file(13) with time information
nlost	(D=10)	maximum number of lost particles
igerr	(D=0)	maximum number of recovery of region error
igchk	(D=1)	flight mesh after region crossing is set by deltb
deltb	(D=1.e-5)	flight mesh after region crossing (cm)
deltm	(D=20.0)	max flight mesh (cm)
delt0	(D=0.1)	minimum flight mesh in nspred, imagnf (cm)
imout	(D=0)	display option for material in input echo
	= 0	208Pb type
	= 1	Pb-208 type
	= 2	82208.0 type
ivout	(D=0)	display option for volume input echo
	= 0	write in [volume] section
	= 1	write in [region]
ipout	(D=1)	display option for importance input echo
	= 0	write in [importance] section
	= 1	write in [region] section
file(2)	(D=cgview.in)	output file name of CGVIEW input when icntl=2
file(3)	(D=cgview.set)	setup file name for CGVIEW input when icntl=2
file(4)	(D=marspf.in)	output file name for MARS-PF input when icntl=4
file(6)	(D=non)	file name of summary of the calculation
		if not specified, write on standard output unit 6
file(12)	(D=fort.12)	output file name of cutoff neutron
file(13)	(D=fort.13)	output file name of cutoff γ
file(14)	(D=fort.14)	input file name of the data for γ emission for igamma=1
inucr	(D=1)	option for nuclear reaction calculation
	= 1	double differential cross section
	= 2	total, elastic and nonelastic cross section
	= 3	nonelastic cross section in simulation
	= 4	angular distribution of elastic collision
	= 5	particle-particle like pp, np, π-p, cross section
	= 6	particle-particle like pp, np, π-p, cross section in simulation

9.5. [S o u r c e] Section

In this section, the information of the incident particle is defined. The following 8 types of source are available in the code. The type of source is specified by `s-type =` *number*.

Table 12. Source type.

Source type	explanation
`s-type = 1`	Cylinder or circle including pencil beam
`s-type = 2`	Rectangular
`s-type = 3`	Gaussian distribution
`s-type = 4`	Cylinder with energy distribution
`s-type = 5`	Rectangular with energy distribution
`s-type = 11`	Uniform distribution in phase space normal to the beam direction
`s-type = 12`	read the output of DECAY-TURTLE
`s-type = 100`	generated by the user defined source subroutine compile with usrsors.f

The parameters for each source type are shown below.

Table 13. Source parameters (1).

parameter	explanation
`s-type =`	source type (common for any source type)
`proj =`	incident particle, given by ityp or kf-code (see Table 1)
`s-type = 1, 4`	Cylinder
`x0 =`	x-coordinate of the center of cylinder (cm) `(D=0.0)`
`y0 =`	y-coordinate of the center of cylinder (cm) `(D=0.0)`
`z0 =`	minimum z-coordinate of cylinder (cm)
`z1 =`	maximum z-coordinate of cylinder (cm)
`r0 =`	radius of cylinder. Zero is given for pencil beam (cm)
`dir =`	directional cosine to z-axis of incident particle if `dir=all` is given, the direction is isotropic
`phi =`	azimuthal angle of beam direction (degree) if neglected, the azimuthal angle is random
`dom =`	spread of beam direction (degree) `(D=0.0)`
`e0 =`	energy of incident particle (for s-type=1) (MeV)

Table 14. Source parameters (2).

parameter	explanation
s-type = 2, 5	Rectangular
x0 =	minimum x-coordinate of rectangular (cm)
x1 =	maximum x-coordinate of rectangular (cm)
y0 =	minimum y-coordinate of rectangular (cm)
y1 =	maximum y-coordinate of rectangular (cm)
z0 =	minimum z-coordinate of rectangular (cm)
z1 =	maximum z-coordinate of rectangular (cm)
dir =	the directional cosine to z-axis of incident particle if dir=all is given, the direction is isotropic
phi =	the azimuthal angle of beam direction (degree) if neglected, the azimuthal angle is random
dom =	spread of beam direction (degree) (D=0.0)
e0 =	energy of incident particle (for s-type=2) (MeV)
s-type = 3	Gaussian distribution
x0 =	x-coordinate of the center of gaussian (cm)
x1 =	x-width at half maximum value (cm)
y0 =	y-coordinate of the center of gaussian (cm)
y1 =	y-width at half maximum value (cm)
z0 =	z-coordinate of the center of gaussian (cm)
z1 =	z-width at half maximum value (cm)
dir =	the directional cosine to z-axis of incident particle if dir=all is given, the direction is isotropic
phi =	the azimuthal angle of beam direction (degree) if neglected, the azimuthal angle is random
dom =	spread of beam direction (degree) (D=0.0)
e0 =	energy of incident particle (MeV)
s-type = 11	Uniform distribution in phase space normal to the beam direction
x0 =	x-coordinate of the center of beam (cm)
x1 =	ratio of maximum radius and maximum angle in x-direction (cm/mrad)
y0 =	y-coordinate of the center of beam (cm)
y1 =	ratio of maximum radius and maximum angle in x-direction (cm/mrad)
z0 =	minimum z-coordinate (cm)
z1 =	maximum z-coordinate (cm)
rx =	angle in x-direction of the ellipse of phase space (rad)
ry =	angle in y-direction of the ellipse of phase space (rad)
wem =	emittance (π cm $\times$ mrad)
dir =	the directional cosine to z-axis of incident particle
e0 =	energy of incident particle (MeV)
s-type = 12	read output of decay-turtle
x0 =	x-offset of the beam (cm)
y0 =	y-offset of the beam (cm)
z0 =	z-offset of the beam (cm)
dir =	the directional cosine to z-axis of incident particle
file =	file name of output of decay-turtle

The format of the output data of `decay-turtle` is the double precision and ASCII format, and each record is given by

```
xp, xq, yp, yq, e0, wt0, pz0
```

These variables have the following meanings.

Table 15. Data of `decay-turtle`.

variables	explanation
xp, yp	position of the beam particle(cm)
xq, yq	angle of momentum to the beam direction (mrad)
e0	momentum of the beam particle (GeV/c)
wt0	wight of the beam particle
pz0	polarization of the beam particle (not used)

For the source type with energy distribution, `s-type = 4, 5`, the following parameters which define the energy distribution are necessary. There are three type of energy distribution.

Table 16. parameters for energy distribution

parameter	explanation
e-type = 1	give energy mesh and weight
ne =	number of energy group data are given in free format and the data sequence is in FORTRAN form as (e(i),w(i),i=1,ne), e(ne+1)
e-type = 2	Gaussian distribution
eg0 =	center value of gaussian (MeV)
eg1 =	width at half maximum value (MeV)
eg2 =	minimum value of gaussian distribution (cutoff) (MeV)
eg3 =	maximum value of gaussian distribution (cutoff) (MeV)
e-type = 3	Maxwellian distribution : $f(x) = x^{1.5}\exp(-x/T)$
et0 =	Temperature parameter T (MeV)
et1 =	minimum value of Maxwellian distribution (cutoff) (MeV)
et2 =	maximum value of Maxwellian distribution (cutoff) (MeV)

9.6. [M a t e r i a l] Section

In this section, the materials which compose the system are defined. The basic format of this section is

```
[ M a t e r i a l ]
  MAT[  number ]
      nucleus     density
      nucleus     density
      nucleus     density
  MAT[  number ]
      nucleus     density
      ......      .......
      ......      .......
```

The number of material can be used from 1 to 9999 as long as it is not duplicate. A typical example of the material section is the following.

List 9.6 ● material example (1)

```
 1:  [ M a t e r i a l ]
 2:  MAT[ 1 ]
 3:     1H         1.0000000E-04
 4:   208Pb        1.7238000E-02
 5:   204Pb        4.6801000E-04
 6:   206Pb        7.9430000E-03
 7:   207Pb        7.2838000E-03
 8:  MAT[ 2 ]
 9:     1H         1.0000000E-09
10:    14N         4.6801000E-05
11:    16O         7.9430000E-06
```

If you change the order of data, *nucleus* and *density*, you should put the data sequence identification explicitly as

List 9.7 ● material example (2)

```
 1:  [ M a t e r i a l ]
 2:     den             nuc       <------
 3:  MAT[ 1 ]
 4:  1.0000000E-04       1H
 5:  1.7238000E-02     208Pb
 6:  4.6801000E-04     204Pb
 7:  7.9430000E-03     206Pb
 8:  7.2838000E-03     207Pb
 9:  MAT[ 2 ]
10:  1.0000000E-09       1H
11:  4.6801000E-05      14N
12:  7.9430000E-06      16O
```

where **den**, and **nuc** in 2nd line are the data sequence identification. There are three format to describe the nucleus, `208Pb, Pb-208, 82208.0`. Any type of format can be read. The density should be given in [10^{24} `atoms/cm`^{3}] unit.

9.7. [B o d y] Section

In this section, the body of geometry configuration is defined. After the title of the body section, a comment can be written for the output of CG package. In the first part of the body section, three parameters should be defined, **idbg, ibod, naz**. These can be omitted, in that case, the default values are set.

Table 17. control parameters for body.

parameter	value	explanation
idbg	(D=0)	option for debug
	= 0	no debug output
	= 1	write input echo for CG
	= 2	write debug output for CG
ibod	(D=1)	option for body number
	= 0	omit body number
	> 0	body number is given
naz	(D=0)	Number of regions to be added to the data storage for next region of entry memory table. Enter any large number if extra storage is required. (give 0 usually)

These parameters are written in one line like,

```
idbg = 0 ;  ibod = 1 ;  naz =  0
```

The data which define the body are the name of body (**sym**), number of body (**num**), and the geometry configuration data (**def**). The data sequence of these three data are specified by the data sequence identification symbol **sym**, **num**, and **def**, while **def** should be always the last position. If there are data columns which you want to skip, the symbol **non** is used for the columns. The default data sequence is

```
num   sym   def
```

In the case without the body number, the code assigns the sequential number for the body from 1. The data of geometry configuration is read automatically from some sequential lines. Four typical examples are given in the followings.

List 9.8 ● body example (1)

```
[ B o d y ]   body example 1
     1   rpp    -7.5000000E+00  7.5000000E+00
                -7.5000000E+00  7.5000000E+00
                -1.0000000E+01  1.0000000E+01
     2   sph     0.0000000E+00  0.0000000E+00  0.0000000E+00  9.9900000E+01
     3   sph     0.0000000E+00  0.0000000E+00  0.0000000E+00  1.0000000E+02
```

List 9.9 ● body example (2)

```
[ B o d y ]   body example 2
 idbg = 0 ;  ibod = 0 ;  naz =  0
   rpp    -7.5000000E+00  7.5000000E+00
          -7.5000000E+00  7.5000000E+00
          -1.0000000E+01  1.0000000E+01
   sph     0.0000000E+00  0.0000000E+00  0.0000000E+00  9.9900000E+01
   sph     0.0000000E+00  0.0000000E+00  0.0000000E+00  1.0000000E+02
```

List 9.10 ● body example (3)

```
[ B o d y ]   body example 3
 idbg = 0 ;  ibod = 1 ;  naz =  0
  num   sym     def
    1   rpp    -7.5000000E+00  7.5000000E+00
               -7.5000000E+00  7.5000000E+00
               -1.0000000E+01  1.0000000E+01
    2   sph     0.0000000E+00  0.0000000E+00  0.0000000E+00  9.9900000E+01
    3   sph     0.0000000E+00  0.0000000E+00  0.0000000E+00  1.0000000E+02
```

List 9.11 ● body example (4)

```
[ B o d y ]   body example 4
 idbg = 0 ;  ibod = 1 ;  naz =  0
  non   sym  num  def
    1   rpp   5  -7.5000000E+00  7.5000000E+00 -7.5000000E+00  7.5000000E+00
                 -1.0000000E+01  1.0000000E+01
    2   sph   7   0.0000000E+00  0.0000000E+00  0.0000000E+00  9.9900000E+01
    3   sph   8   0.0000000E+00  0.0000000E+00  0.0000000E+00  1.0000000E+02
```

In the 4th example, the body number is read from third column and the first column is skipped.

9.8. [R e g i o n] Section

In this section, the region of geometry configuration, importance (if required), universe(for the use of array), and volume(if required) are defined. The data which should be given are the region number (num), material number (mat), region symbol (sym), and definition of region (def), and if required, volume (vol), importance (imp) and universe (uni).

The negative material numbers refers a valid array number which absolute value corresponds to the array number. The following material numbers have special meanings, i.e. 0 means outer void, 1000 is inner void and -1000 represents outer boundary of universe. The region symbol is given within three characters whose first one is given by an alphabet.

These data sequence is specified and changed by the data sequence identification symbol, num, mat, sym, def, vol, imp and uni, while def should be always the last position. In the def, the equations nor the user defined constants cannot be used. The default sequence is

```
num   mat  sym    def
```

If there are data columns which you want to skip, the symbol non is used for the columns. The data of geometry configuration is read automatically from some sequential lines. Three examples are given in the followings.

List 9.12 ● region example (1)

```
1:  [ R e g i o n ]
2:      1      1   tgt    +1
3:      2      2   iA5    -2 +3 +6 -7
4:      3      2   oA5    -3 +4 +6 -7
5:      4      2   iA2    -2 +3 +8 -9
6:      5      2   oA2    -3 +4 +8 -9
```

List 9.13 ● region example (2)

```
1:  [ R e g i o n ]
2:    num    mat   imp          vol          sym       def
3:      1      1   1.000000     1.000000     tgt       +1
4:      2      2   2.000000     2.000000     iA5       -2 +3 +6 -7
5:      3      2   4.000000     1.000000     oA5       -3 +4 +6 -7
6:      4      2   8.000000     1.000000     iA2       -2 +3 +8 -9
7:      5      2   16.00000     3.000000     oA2       -3 +4 +8 -9
```

List 9.14 ● region example (3)

```
1:  [ R e g i o n ]
2:    num    mat   non          non      sym       def
3:      1      1   1.000000     0.0      tgt       +1
4:      2      2   2.000000     0.0      iA5       -2 +3 +6 -7
5:      3      2   4.000000     0.0      oA5       -3 +4 +6 -7
6:      4      2   8.000000     0.0      iA2       -2 +3 +8 -9
7:      5      2   16.00000     0.0      oA2       -3 +4 +8 -9
```

Example (1) and example (3) are equivalent. In the example (2), if you want to neglect the importance, you can replace imp by non.

9.9. [A r r a y] Section

In this section, the array of geometry configuration is defined. The description of this section is, exceptionally, in card image.

- CARD 1 (Free)

  (1) **NXMAX**

  The length of an array along x-direction. If array is not used, zero must be given.

  (2) **NYMAX**

  The length of an array along y-direction. If array is not used, zero must be given.

  (3) **NZMAX**

  The length of an array along z-direction. If array is not used, zero must be given.

  An array is a regular rectangular lattice composed of rectangular cells of arbitrary content. The size of array should be entered as **NXMAX**, **NYMAX** by **NZMAX**. Arrays are sequentially labelled from 1 as they are entered. The array size entered should include any vacant cells in the array, if any are present. After the size of the last array has been entered, a zero should be entered to terminate the entries. Zero is an illegal entry for array size.

- CARD 2 (Free)

  (1) IOP

  Option for Array specification.
  = 0 : Free-Form FIDO style
  = 1 : Do loop style, (the same as that adopted in KENO code).
  = 2 : Standard FIDO style, (the same as that adopted in ANISN code).

- CARD 3A (Free) [IOP = 0]

  (1) NCA(1:NXMAX, 1:NYMAX, 1:NZMAX, 1:NAR)

  The contents of each cell of each array. The number of cells along x-direction, **NXMAX**, should be given at first. Then, that along y-direction, **NYMAX** and that along z-direction, **NZMAX**, are given, respectively. Those data must be required to repeat **NAR** times. Here, if a positive value is given, the cell is universe. The negative one means that the cell is array. When zero is supplied, the cell is empty.

- CARD 3B (Free) [IOP = 1]

  (1) LTYPE

  The number of a cell. The positive, negative and zero values indicate that the cell is universe, array and empty cell, respectively.

(2) IX1

The starting point of the cell LTYPE in the x-direction. This value must be at least 1 and less than or equal to NXMAX.

(3) IX2

The ending point of the cell LTYPE in the x-direction. This value must be at least 1 and less than or equal to NXMAX.

(4) INCX

The number of cells by which increments are made in the positive x-direction. INCX must be greater than zero and less than or equal to NXMAX.

(5) IY1

The starting point of the cell LTYPE in the y-direction. This value must be at least 1 and less than or equal to NYMAX.

(6) IY2

The ending point of the cell LTYPE in the y-direction. This value must be at least 1 and less than or equal to NYMAX.

(7) INCY

The number of cells by which increments are made in the positive y-direction. INCY must be greater than zero and less than or equal to NYMAX.

(8) IZ1

The starting point of the cell LTYPE in the z-direction. This value must be at least 1 and less than or equal to NZMAX.

(9) IZ2

The ending point of the cell LTYPE in the z-direction. This value must be at least 1 and less than or equal to NZMAX.

(10) INCZ

The number of cells by which increments are made in the positive z-direction. INCZ must be greater than zero and less than or equal to NZMAX.

(11) INTP

$= 0$: read another set of data.

$\neq 0$: do not read any more mixed-cell orientation data.

The data sets IX1, IX2, INCX and so on from (2) to (10) correspond to the variables which specify the starting value, ending one and increments of do-loop statement in the FORTRAN language. An important feature of this type of data description is that, if any portion of an array is defined in a conflicting manner, the last card to define that portion will be the one that determines the array's cell type configuration. To utilize this feature, one can superimpose the other cell types in their proper places to

accurately describe the array. The last set of mixed-cell orientation data must have a non-zero entry in the last field.

- CARD 3C (Free) [IOP = 2]

  The description for each lattice array is entered as a single array block with FIDO. The FIDO integer array number is the array number being described plus 100. Since the array being entered is integer, it is a "$" or "$$" array. The data is entered and each array description is terminated with "T". Thus, array 1 would be entered as the "101$$". All standard FIDO repeat options are available for entering the data. The format for the data entry is the same as the description for free-form input. All x entries for the first y row and first z level are entered, then all x entries for the second row and first z level are entered. This process continues until the entire first z level has been described. Then, the second z level is described until the entire array has been described. Then, the geometry array description or a given array is terminated with a "T".

- CARD 4 (Free)

  (1) ITU(1:N)

  Type of universe. The entries should be either zero or 1.

  = 0 : Universe is "combinatorial".

  = 1 : Universe is "simple".

  A "simple" universe is a universe composed of concentric zones, where every zone completely surrounds the zone inside of it. Furthermore, input zones in a simple universe may be only one code zone and may be described by only one or two bodies. Figure 47 and 48 shows the sketches of a simple and a combinatorial universe.

9.10. [I m p o r t a n c e] Section

In this section, the importance of the region is defined. The importance is also defined in [Region] section. If the importance is defined in both sections, the values defined in this section have priority. If the importance is not defined in both sections, the unity is set for the region.

The basic format of this section is

```
[ I m p o r t a n c e ]
   reg        imp
     1      1.000000
     2      2.000000
     3      3.000000
   ....     ........
   ....     ........
```

If you want to exchange the data sequence of the region number (**reg**) and the importance (**imp**), you write " **imp reg** ". The data identification of "non" can be used for the column which you want to skip.

9.11. [V o l u m e] Section

In this section, the volume of the region is defined. The unit is cm^3. The volume is also defined in [Region] section. If the volume is defined in both sections, the values defined in this section have priority. If the volume is not defined in both sections, 1.0 cm^3 is set for the region.

The basic format of this section is

```
[ V o l u m e ]
   reg        vol
     1      1.000000
     2      2.000000
     3      3.000000
   ....     ........
   ....     ........
```

If you want to exchange the data sequence of the region number (**reg**) and the volume (**vol**), you write " **vol reg** ". The data identification of "non" can be used for the column which you want to skip.

9.12. [M a g n e t i c F i e l d] Section

In this section, the magnetic field of the region is defined. The data which should be given are the region number (**reg**), the type of the magnetic field (**typ**), the half gap of the magnet (**gap**) and the strength of the magnetic field (**mgf**). These data sequences are specified and changed by the data sequence identification symbol, **reg**, **typ**, **gap** and **mgf**. You can use also "non" for the skip columns. The type of the magnetic field is defined by "2" for dipole case, "4" for quadrupole case. The unit of gap is [cm] and [kG] for the strength.

The basic format of this section is

```
[ M a g n e t i c  F i e l d ]
  reg    typ    gap           mgf
    1      4    10.00000     -5.956540
    2      4    10.00000      6.416140
    3      2    10.00000     -7.611980
    4      2    10.00000      3.516000
   ...    ...   ........      ........
   ...    ...   ........      ........
```

For the dipole case, a certain number should be put on the "**gap**" column, though the gap has no meaning.

9.13. Geometry Mesh for Tallies

There are six tally sections as,

Table 18. List of tally sections.

name	explanation
[t-track]	defines the parameters of track length tally
[t-cross]	defines the parameters of surface crossing tally
[t-yield]	defines the parameters of nuclide yield tally
[t-heat]	defines the parameters of heat tally
[t-time]	defines the parameters of time tally
[t-star]	defines the parameters of star density tally

In the following, we explain the descriptions commonly used in the tally sections.

In each tally section, the user can use three kinds of geometry mesh, the region mesh (**reg**) of CG, r-z scoring mesh (**r-z**) and xyz scoring mesh (**xyz**). The format of the definition of the geometry mesh in the tally section is

```
mesh =   [ reg,  r-z,  xyz ]
```

9.13.1 Region Mesh

For the region mesh case, the region is selected by

```
mesh = reg
 reg = 1 2 3 4 5 10 11 50
```

In the right-hand side, you put the region numbers separated by blanks. For sequential region numbers, the following expression is also available.

```
mesh = reg
 reg = { 1 - 5 } 10 11 50
```

In this { n1 - n2 } expression, n1 < n2 should be satisfied. If you want to select all region for the tally, you can use "all" as

```
mesh = reg
 reg = all
```

9.13.2 r-z Mesh

For the r-z scoring mesh, you can set the offset of the center of r-z mesh in x-y coordinate as

```
mesh = r-z
  x0 = 1.0
  y0 = 2.0
```

This offset definition can be omitted. For that case, the default values, x0 = 0.0 and y0 =0.0, are set for the offset. The values of meshes of r and z are defined by the subsection of mesh definition. The subsection of mesh definition is started from the type definition as

```
mesh   = r-z
r-type = [1-5]
   ..........
   ..........
z-type = [1-5]
   ..........
   ..........
```

The detail of the subsection of mesh definition is described later.

9.13.3 xyz Mesh

For the xyz scoring mesh, the values of meshes of x, y and z are defined by the subsection of mesh definition. The subsection of mesh definition is started from the type definition as

```
mesh   = xyz
x-type = [1-5]
   ..........
   ..........
y-type = [1-5]
   ..........
   ..........
z-type = [1-5]
   ..........
   ..........
```

The detail of the subsection of mesh definition is described later.

9.14. Energy Mesh for Tallies

The energy mesh used in tallies is defined by the subsection of mesh definition as

```
e-type = [1-5]
   ..........
   ..........
```

The detail of the subsection of mesh definition is described later.

9.15. Time Mesh for Tallies

The time mesh used in tallies is defined by the subsection of mesh definition as

```
t-type = [1-5]
   ..........
   ..........
```

The detail of the subsection of mesh definition is described later.

9.16. Mesh Definition Subsection for Tallies

There are six kinds of mesh definition subsection starting from `e-type`, `t-type`, `x-type`, `y-type`, `z-type`, `r-type`. Since these descriptions are the same except for the first character,

we explain here the case of **e-type**. Then you exchange, for example, "**ne**" by "**nx, ny, nz, ...**" or "**emin**" by "**xmin, ymin, zmin, ...**" for the other quantities.

9.16.1 Mesh Type

There are five types to define the number of mesh group and mesh points, which are selected by

```
e-type = [1-5]
```

The mesh types are summarized in the following table.

Table 19. Mesh type.

mesh type	explanation
1	number of group and mesh points are given by the data
2	number of group, minimum and maximum values are given by the data and generate mesh points by a equal width in linear scale
3	number of group, minimum and maximum values are given by the data and generate mesh points by a equal width in logarithmic scale
4	width of mesh, minimum and maximum values are given by the data and generate mesh points by a equal width in linear scale and number of group
5	width of mesh, minimum and maximum values are given by the data and generate mesh points by a equal width in logarithmic scale and number of group

9.16.2 e-type = 1

```
e-type = 1
    ne = number of group
         data(1) data(2) data(3) data(4)
         data(5) data(6) data(7) data(8)
         .........
         .........
         data(ne+1)
```

9.16.3 e-type = 2, 3

```
e-type = 2, 3
    ne = number of group
  emin = minimum value
  emax = maximum value
```

9.16.4 e-type = 4

```
e-type = 4
  edel =  width of mesh
  emin =  minimum value
  emax =  maximum value
```

9.16.5 e-type = 5

```
e-type = 5
  edel =  width of mesh
  emin =  minimum value
  emax =  maximum value
```

In this case, the width of mesh is that in logarithmic scale, i.e. edel = log(E_{i+1} / E_i).

9.17. Other Definitions for Tallies

9.17.1 Particle Definition

The definition of the particles for which quantities are scored in the tally is given by

```
part = proton neutron pion+
```

In the above expression, the particles are written in a line by the blank separators. Another definition is

```
part = proton
part = neutron
part = pion+
```

The symbols of the particle are listed in Table 1. Instead of the symbols, the kf-code is also available to define the particles. If you use "all" for the particle name like

```
part = all
```

This means the sum of all particles. In a tally section, you can define maximum six particles. If you want to use more than six particles, you must define another tally section.

9.17.2 Axis Definition

The user can define the horizontal axis of the data in the output of tally. The following axis identifications are prepared, which depend on the kind of tally and the geometry mesh of tally.

```
eng,  reg,  x,  y,  z,  r,  t,  xy,  yz,  zx,  rz,
mass,  charge,  chart,  dchain
```

These are defined as

```
axis = eng
```

In a tally section, you can define some axes like

```
axis = eng  x  y
```

or

```
axis = eng
axis = x
axis = y
```

For each axis, the output is written on a file. Therefore, you should define the same number of axis as the output file.

9.17.3 File Definition

The output file name is defined as

```
file = file.001  file.002  file.003
```

or

```
file = file.001
file = file.002
file = file.003
```

9.17.4 Unit Definition

The format is

```
unit = number
```

The meaning of the number and unit are discussed in each tall sub-section.

9.17.5 Output Definition

The format is

```
output = name of output
```

The meaning of the name of output is discussed in each tall sub-section.

9.17.6 Info Definition

The format is

```
info = 0, 1
```

This is an option whether the detail information is printed out (1) or not (0) in the output of tally.

9.17.7 Title Definition

The format is

```
title =  title of the tally
```

This is a title of the tally. This definition can be omitted, in this case, the default title is printed out on the output file.

9.17.8 *ANGEL* Parameter Definition

The format is

```
angel = xmin(1.0) ymin(1.3e-8)
```

This parameters for *ANGEL* are added to the output of tally. In the output, this is written like

```
p: xmin(1.0) ymin(1.3e-8)
```

The usage of *ANGEL* and the other detail of *ANGEL* are presented in user's manual of *ANGEL* [58].

9.17.9 2d-type Definition

The format is

```
2d-type =  1, 2, 3, 4, 5
```

This parameter is an option of the output format of two-dimensional data when you choose the two-dimensional axis like `axis = xy`.

- `2d-type = 1, 2, 3`
  The format of the output data is expressed in FORTRAN format,

  ```
  ( ( data(ix,iy), ix = 1, nx ), iy = ny, 1, -1 )
  ```

  In a line, there are ten data with *ANGEL* header. The header is for the contour plot by 1, the cluster plot by 2, and the color cluster plot by 3.

- `2d-type = 4`
  In this case, the data sequence is expressed in FORTRAN format,

```
do iy = ny, 1, -1
do ix = 1, nx
   ( x(ix), y(iy), data(ix,iy) )
end do
end do
```

In a line, there are three data, `x(ix)`, `y(iy)`, `data(ix,iy)` , or for the case that some particles are specified,

```
do iy = ny, 1, -1
do ix = 1, nx
   ( ( x(ix), y(iy), data(ix,iy,ip) ), ip = 1, np )
end do
end do
```

where `np` is the number of particles.

- `2d-type = 5`

  In this case, the data sequence is expressed in FORTRAN format,

```
y/x  ( x(ix), ix = 1, nx )
  do iy = ny, 1, -1
     ( y(iy), data(ix,iy), ix = 1, nx )
  end do
```

  In a line, there are `nx + 1` data, totally `ny + 1` lines. This format is appropriate for the other applications like Excel.

9.18. [T - T r a c k] Section

This section defines the parameters of the track length tally. The parameters are shown in the following table.

Table 20. [t-track] parameters.

name	value	explanation
mesh =	reg, r-z, xyz	geometry mesh geometry mesh subsection is necessary
part =	*particle name*, (D=all)	maximum six particles
e-type =	1, 2, 3, 4, 5	energy mesh energy mesh subsection is necessary
unit =	1, 2, 3	1: [1/cm^2/source] 2: [1/cm^2/MeV/source] 3: [1/cm^2/Lethargy/source]
axis =	eng, reg, x, y, z, r xy, yz, xz, rz	horizontal axis of output two dimensional output
file =	*file name*	output file name with path
title =		title of this tally
angel =		*ANGEL* parameter
2d-type =	1, 2, 3, 4, 5	option for 2d data format

9.19. [T - C r o s s] Section

This section defines the parameters of the surface crossing tally. The parameters are shown in the following table.

Table 21. [t-cross] parameters.

name	value	explanation
mesh =	reg, r-z, xyz	geometry mesh geometry mesh subsection is necessary
part =	*particle name*, (D=all)	maximum six particles
e-type =	1, 2, 3, 4, 5	energy mesh energy mesh subsection is necessary
unit =	1, 2, 3	1: [1/cm^2/source] 2: [1/cm^2/MeV/source] 3: [1/cm^2/Lethargy/source]
axis =	eng, reg, x, y, z, r xy	horizontal axis of output two dimensional output
file =	*file name*	output file name with path
output =	flux current f-curr b-curr o-curr of-curr ob-curr	print the flux print the net current print the forward current print the backward current print the omni net current omni means energy integrated quantity print the omni forward current print the omni backward current
title =		title of this tally
angel =		*ANGEL* parameter
2d-type =	1, 2, 3, 4, 5	option for 2d data format

For the **mesh = reg** case, the subsection of the geometry mesh is different from in the other tallies. Two regions are necessary to define the crossing surface. The format is

```
mesh = reg
 reg =  number of crossing surfaces
  r-in     r-out     area
    2        8       10.0
    3        8        5.0
   ...      ...      ....
   ...      ...      ....
```

The data sequence is specified by the data identification symbol, **r-in, r-out, area**. **r-in** and **r-out** are in-coming and out-going regions which define the crossing surface. The unit of area is cm^2.

For the **mesh = r-z** case, z-crossing surfaces and r-crossing surfaces are defined. The former is specified by $r_i - r_{i+1}$ and z_j, total $nz + 1$ surfaces, while the latter is specified by $z_i - z_{i+1}$ and r_j, total $nr + 1$ surfaces.

For the **mesh = xyz** case, only z-crossing surfaces are defined, specified by $x_i - x_{i+1}$, $y_j - y_{j+1}$ and z_k, total $nz + 1$ surfaces.

9.20. [T - Y i e l d] Section

This section defines the parameters of the nuclide yield tally. The parameters are shown in the following table.

Table 22. [t-yield] parameters.

name	value	explanation
mesh =	reg, r-z, xyz	geometry mesh geometry mesh subsection is necessary
special =	*number*, (D=0)	nuclear reaction is repeated *number* times. This increases the statistics of the yield tally
mother = (next lines)	*number*, (D=all) 208Pb Fe	specify the mother nuclei. If neglected or "all" case, all mothers are included. In the sequential lines, mother nuclei are described. Without mass number like Fe, all isotopes are chosen.
nucleus = (next lines)	*number*, (D=all) 208Pb Fe	specify the output nuclei. If neglected or "all" case, all nuclei are printed. In the sequential lines, output nuclei are described. Without mass number like Fe, all isotopes are chosen.
unit =	1, 2	1: [1/source] 2: [1/cm^3/source]
axis =	eng, reg, x, y, z, r xy, yz, xz, rz mass charge chart dchain	horizontal axis of output two dimensional output print mass distribution. with nucleus =, print isotope distribution print charge distribution for this case, nucleus = cannot be specified print 2d data in nucleus chart format horizontal axis is neutron number, vertical axis is charge number. for this case, nucleus = cannot be specified print data for dchain-sp. all isotope are printed. only for mesh = reg
file =	*file name*	output file name with path
info =	0, 1 (D=0)	for dchain, the error is printed on the other file for chart, stable nuclei and magic number are printed.
title =		title of this tally
angel =		*ANGEL* parameter
2d-type =	1, 2, 3, 4, 5	option for 2d data format

9.21. [T - H e a t] Section

This section defines the parameters of the heat tally. The parameters are shown in the following table.

Table 23. [t-heat] parameters.

name	value	explanation
mesh =	reg, r-z, xyz	geometry mesh geometry mesh subsection is necessary
axis =	reg, x, y, z, r xy, yz, xz, rz	horizontal axis of output two dimensional output
file =	*file name*	output file name with path
output =	heat simple all	total heat, cutoff neutron energy, cutoff γ energy and leakage energy are printed. addition to the above, the components of heat, recoil, ionization and the other energy are printed. addition to the above, recoil energy of d, t, ^{3}He, α, and residual nucleus, ionization energy of proton, π^+, π^- and the other particles, and remaining excitation energy, high energy γ, cutoff proton, π^+, π^- and the other stopped particles energy, and the component of fusion are printed.
part =	*particle name*	for the particles, ionization energy and stopped energy are printed.
unit =	1, 2	1: [MeV/cm^3/source] 2: [MeV/source]
title =		title of this tally
angel =		*ANGEL* parameter
2d-type =	1, 2, 3, 4, 5	option for 2d data format

9.22. [T - S t a r] Section

This section defines the parameters of the star density tally. The parameters are shown in the following table.

Table 24. [t-star] parameters.

name	value	explanation
mesh =	reg, r-z, xyz	geometry mesh geometry mesh subsection is necessary
part =	*particle name*, (D=all)	maximum six particles
mother = (next lines)	*number*, (D=all) 208Pb Fe	specify the mother nuclei. If neglected or "all" case, all mothers are included. In the sequential lines, mother nuclei are described. Without mass number like Fe, all isotopes are chosen.
e-type =	1, 2, 3, 4, 5	energy mesh energy mesh subsection is necessary
unit =	1, 2	1: [1/cm^3/source] 2: [1/cm^3/MeV/source]
axis =	eng, reg, x, y, z, r xy	horizontal axis of output two dimensional output
file =	*file name*	output file name with path
output =	all decay elastic nuclear fission absorption	all reactions decay reactions elastic reactions non-elastic and Hydrogen reactions fission reactions absorption reactions
title =		title of this tally
angel =		*ANGEL* parameter
2d-type =	1, 2, 3, 4, 5	option for 2d data format

9.23. [T - T i m e] Section

This section defines the parameters of the time tally. The parameters are shown in the following table.

Table 25. [t-star] parameters.

name	value	explanation
mesh =	reg, r-z, xyz	geometry mesh geometry mesh subsection is necessary
part =	*particle name*, (D=all)	maximum six particles
t-type =	1, 2, 3, 4, 5	time mesh time mesh subsection is necessary
e-type =	1, 2, 3, 4, 5	energy mesh energy mesh subsection is necessary
unit =	1, 2, 3, 4	1: [1/source] 2: [1/nsec/source] 3: [1/nsec/cm^3/source] 4: [1/nsec/cm^3/MeV/source]
axis =	t, eng, reg, x, y, z, r xy, yz, xz, rz	horizontal axis of output two dimensional output
file =	*file name*	output file name with path
output =	all cutoff escape	energy cutoff and escape particles energy cutoff particles escape particles
title =		title of this tally
angel =		*ANGEL* parameter
2d-type =	1, 2, 3, 4, 5	option for 2d data format

9.24. Data Format of History Files for Cutoff Neutron and Photon

As for the cutoff neutrons and photons below the cutoff energy defined by the user, emin(2) and emin(14), the phase space information of these cutoff particles can be stored in the file for sequential calculations. The transport of the low energy neutrons and photons can be calculated by the neutron-photon transport code such as MCNP using a cross section library processed from evaluated nuclear data.

The data are written on the files, "`file(12)`" and "`file(13)`" for neutron and photon respectively, defined by the user with the binary format. The data sequence of cutoff neutrons in the history file is shown in the followings, which are the same as for the photon case.

```
rd, rn, ( data(i), i = 1, nint(abs(rd)) )
rd, rn, ( data(i), i = 1, nint(abs(rd)) )
.................
.................
```

The above expression is rewritten for each cases in terms of the each variables.
For the case of `incut = 1` without the importance (rd>0),

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), i = 1, n )
.................
.................
```

For the case of `incut = 1` with the importance (rd<0),

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), i = 1, n )
.................
.................
```

For the case of `incut = 2` without the importance (rd>0),

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), t(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), t(i), i = 1, n )
.................
.................
```

For the case of incut = 2 with the importance (rd<0),

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), t(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), t(i), i = 1, n )
.................
.................
```

In the above expressions, n = nint(rn), x, y, z are the coordinate in (cm), e(i) is the energy (MeV), u(i), v(i), w(i) are the unit vector of the momentum, wt(i) is the weight of the particle, and t(i) is the time of the particle (ns).

10. Examples of Execution of NMTC/JAM

10.1. The Simplest Input

Let us start from the simplest input as below.

List 10.1 ● Input example (1)

```
 1:  [ T i t l e ]
 2:  Test Calculation for Example (1)
 3:
 4:  [ P a r a m e t e r s ]
 5:   maxcas  = 10
 6:   maxbch  = 10
 7:   incut   = 0
 8:   igcut   = 0
 9:   file(6) = nmtcjam.out
10:
11:  [ S o u r c e ]
12:   s-type =   1
13:     proj =   proton
14:       e0 =   50000.0
15:       r0 =   1.5000
16:       z0 =   0.0
17:       z1 =   0.0
18:      dir =   1.0000
19:
20:  [ M a t e r i a l ]
21:  MAT[ 1 ]
22:   196Hg        5.9595000E-05
23:   198Hg        3.9611000E-03
24:   199Hg        6.7025000E-03
25:   200Hg        9.1776000E-03
26:   201Hg        5.2364000E-03
27:   202Hg        1.1863000E-02
28:   204Hg        2.2795000E-03
29:  MAT[ 2 ]
30:   204Pb        4.6801000E-04
31:   206Pb        7.9430000E-03
32:   207Pb        7.2838000E-03
33:   208Pb        1.7238000E-02
34:  MAT[ 3 ]
35:    54Fe        4.9161000E-03
36:    56Fe        7.7741000E-02
37:    57Fe        1.8647000E-03
38:    58Fe        2.3733000E-04
39:  MAT[ 4 ]
40:    14N         4.6801000E-05
41:    16O         7.9430000E-06
42:
43:  [ B o d y ]
44:    num   sym    def
45:      1   rcc    0 0  5 0 0 50  5
46:      2   rcc    0 0 10 0 0 45 10
47:      3   rcc    0 0 15 0 0 40 15
48:      4   rcc    0 0  0 0 0 60 20
49:
50:  [ R e g i o n ]
51:    num    mat     sym     def
52:      1      1      Hg      +1
53:      2      2      Pb      +2 -1
```

```
54:      3      3      Fe      +3 -2
55:      4      4     Air      +4 -3 -2 -1
56:      5      0     out      -4
57:
58:  [END]
```

The geometry configuration of the above input is shown in Figure 49.

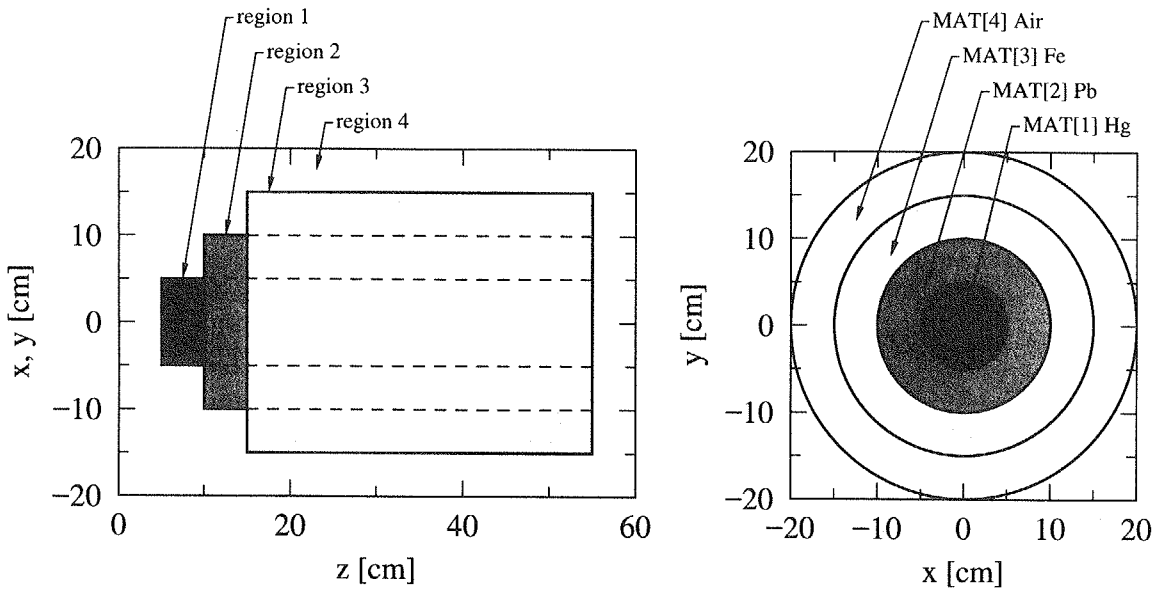

Figure 49. Geometry Configuration of Input Example (1).

In the above input file, we define only 5 parameters in [parameters] section. The other parameters are set by the default values. By `incut = 0, igcut = 0`, we hinder printing the phase-space information of cutoff neutron and photons. The standard output is written in file "`nmtcjam.out`".

10.1.1 Usage of Include File

At any place of the input file, you can include the other file or part of the other file as explained in Section 9.2.3. If you store the each material data in the data file, the material section can be expressed as

List 10.2 ● Example of Include file

```
1:  [ M a t e r i a l ]
2:  MAT[ 1 ]
3:  infl: {Hg.dat}
4:  MAT[ 2 ]
5:  infl: {Pb.dat}
6:  MAT[ 3 ]
7:  infl: {Fe.dat}
8:  MAT[ 4 ]
9:  infl: {Air.dat}
```

In the file, `Hg.dat, Pb.dat, Fe.dat, Air.dat`, the material data are stored as,

List 10.3 ● Data File of Hg.dat

```
1:  196Hg       5.9595000E-05
2:  198Hg       3.9611000E-03
3:  199Hg       6.7025000E-03
4:  200Hg       9.1776000E-03
5:  201Hg       5.2364000E-03
6:  202Hg       1.1863000E-02
7:  204Hg       2.2795000E-03
```

List 10.4 ● Data File of Pb.dat

```
1:   204Pb      4.6801000E-04
2:   206Pb      7.9430000E-03
3:   207Pb      7.2838000E-03
5:   208Pb      1.7238000E-02
```

List 10.5 ● Data File of Fe.dat

```
1:    54Fe      4.9161000E-03
2:    56Fe      7.7741000E-02
3:    57Fe      1.8647000E-03
4:    58Fe      2.3733000E-04
```

List 10.6 ● Data File of Air.dat

```
1:    14N       4.6801000E-05
2:    16O       7.9430000E-06
```

10.2. The Standard Output

The standard output of the execution of NMTC/JAM with above input is shown by part by part in the followings.

10.2.1 Header Part

In the header part of the standard output of NMTC/JAM, the logo and the information of version and date of last revision are shown. In addition, the title of calculation and the date of execution are also printed out.

List 10.7 ● Standard Output List (1)

```
 1:        ------------------------------------------------------------
 2:       |                                                            |
 3:       | NN   NN MM   MM TTTTTTT  CCCCC      //   JJ   AAA   MM   MM |
 4:       | NNN  NN MMM MMM TTTTTTT CCC  CC    //    JJ  AA AA  MMM MMM |
 5:       | NNNN NN MMMMMMMM   TTT   CC        //     JJ AA   AA MMMMMMM |
 6:       | NN NNNN MM M MM   TTT   CC       // JJ   JJ AAAAAAA MM M MM |
 7:       | NN  NNN MM   MM   TTT   CCC  CC //  JJJ JJJ AA   AA MM   MM |
 8:       | NN   NN MM   MM   TTT    CCCCC //    JJJJJ  AA   AA MM   MM |
 9:       |                                                            |
10:       |            Nucleon Meson and JAM Transport Code            |
11:       |                  for High Energy Particles                 |
12:       |                      Version = 1.03                        |
13:       |                         made by                            |
14:       |           Japan Atomic Energy Research Institute           |
15:       |                Last Revised  2000-12-11                    |
16:       |____________________________________________________________|
17:
18:                    ___________ [ Job Title ] __________
19:                   |                                    |
20:                   |  Test Calculation for Example (1)  |
21:                   |____________________________________|
22:
23:                        Starting Date = 2000-12-21
24:                        Starting Time = 15h 29m 30
25:
```

10.2.2 Input Echo Part

The output list from the line 26 to 90 shows the input echo with some comments. By this input echo, you can check your input values of the parameters and its meanings. This part, furthermore, the output file itself, can be used as a input file for the same execution of NMTC/JAM.

List 10.8 ● Standard Output List (2)

```
26:  >>> Input Echo >>>================================================================
27:
28:  [ T i t l e ]
29:  Test Calculation for Example (1)
30:
31:  [ P a r a m e t e r s ]
32:   maxcas   =          10     # (D=10) number of particles per one batch
33:   maxbch   =          10     # (D=10) number of batches
34:   incut    =           0     # (D=1) cutoff neutron on file, 0:No, 2:with Time
35:   igcut    =           0     # (D=1) cutoff gamma on file, 0:No, 2:with Time
36:   file(6)  = nmtcjam.out     # (D=nmtcjam.out) general output file name
37:
38:  [ S o u r c e ]
39:   s-type =   1            # mono-energetic axial source
40:     proj =   proton       # kind of incident particle
41:       e0 =   50000.       # energy of beam [MeV]
42:       r0 =   1.5000       # radius [cm]
43:       z0 =  0.00000E+00   # minimum position of z-axis [cm]
```

```
44:       z1 =  0.00000E+00   # maximum position of z-axis [cm]
45:      dir =   1.0000       # z-direction of beam [cosine]
46:
47:  [ M a t e r i a l ]
48:  MAT[ 1 ]
49:   196Hg       5.9595000E-05
50:   198Hg       3.9611000E-03
51:   199Hg       6.7025000E-03
52:   200Hg       9.1776000E-03
53:   201Hg       5.2364000E-03
54:   202Hg       1.1863000E-02
55:   204Hg       2.2795000E-03
56:  MAT[ 2 ]
57:   204Pb       4.6801000E-04
58:   206Pb       7.9430000E-03
59:   207Pb       7.2838000E-03
60:   208Pb       1.7238000E-02
61:  MAT[ 3 ]
62:    54Fe       4.9161000E-03
63:    56Fe       7.7741000E-02
64:    57Fe       1.8647000E-03
65:    58Fe       2.3733000E-04
66:  MAT[ 4 ]
67:    14N        4.6801000E-05
68:    16O        7.9430000E-06
69:
70:  [ B o d y ]  Body Information
71:   idbg = 0 ;  ibod = 1 ;  naz =  0
72:    num   sym     def
73:      1   rcc     0.0000000E+00  0.0000000E+00  5.0000000E+00
74:                  0.0000000E+00  0.0000000E+00  5.0000000E+01  5.0000000E+00
75:      2   rcc     0.0000000E+00  0.0000000E+00  1.0000000E+01
76:                  0.0000000E+00  0.0000000E+00  4.5000000E+01  1.0000000E+01
77:      3   rcc     0.0000000E+00  0.0000000E+00  1.5000000E+01
78:                  0.0000000E+00  0.0000000E+00  4.0000000E+01  1.5000000E+01
79:      4   rcc     0.0000000E+00  0.0000000E+00  0.0000000E+00
80:                  0.0000000E+00  0.0000000E+00  6.0000000E+01  2.0000000E+01
81:
82:  [ R e g i o n ]
83:    num    mat   imp          sym        def
84:      1      1   1.000000     Hg         +1
85:      2      2   1.000000     Pb         +2 -1
86:      3      3   1.000000     Fe         +3 -2
87:      4      4   1.000000     Air        +4 -3 -2 -1
88:      5      0   1.000000     out        -4
89:
90:  [END] of Input Echo <<<==========================================================
```

10.2.3 Batch Summary Part

Next part is the summary of the batch. In the input, we set 10 batches in List 10.1. Here there are some information of each batch. **ncas** is the total event number until the batch, **rijk** shows the initial random number of the batch. The number of **datalo neutron** is the number of cutoff neutron in a batch. **ncall/s** is the number of calling random number per a source particle. In addition to these quantities, the cpu time is shown for each batch.

List 10.9 ● Standard Output List (4)

```
 91:
 92:  ------------------------------------------------------------------------------
 93:   batch[   1 ]  : ncas =      10  : rijk    =   6647299061401.
 94:         datalo neutron =    4898  : ncall/s =        840523.19
 95:               cpu time =  20.458 s.
 96:
 97:  ------------------------------------------------------------------------------
 98:   batch[   2 ]  : ncas =      20  : rijk    = 148844717761569.
 99:         datalo neutron =    5132  : ncall/s =        886114.38
100:               cpu time =  22.083 s.
101:
102:  ------------------------------------------------------------------------------
103:   batch[   3 ]  : ncas =      30  : rijk    = 159131926320105.
104:         datalo neutron =    5599  : ncall/s =        969461.13
105:               cpu time =  23.734 s.
106:
107:  ------------------------------------------------------------------------------
108:   batch[   4 ]  : ncas =      40  : rijk    = 278364013185009.
109:         datalo neutron =    5339  : ncall/s =        938777.38
110:               cpu time =  22.895 s.
111:
112:  ------------------------------------------------------------------------------
113:   batch[   5 ]  : ncas =      50  : rijk    = 254775167357497.
114:         datalo neutron =    5321  : ncall/s =        937068.50
115:               cpu time =  23.747 s.
116:
117:  ------------------------------------------------------------------------------
118:   batch[   6 ]  : ncas =      60  : rijk    = 100401947972801.
119:         datalo neutron =    5364  : ncall/s =        949891.13
120:               cpu time =  23.688 s.
121:
122:  ------------------------------------------------------------------------------
123:   batch[   7 ]  : ncas =      70  : rijk    = 194872375689609.
124:         datalo neutron =    6201  : ncall/s =       1052812.25
125:               cpu time =  26.588 s.
126:
127:  ------------------------------------------------------------------------------
128:   batch[   8 ]  : ncas =      80  : rijk    = 162780866091665.
129:         datalo neutron =    5159  : ncall/s =        878367.88
130:               cpu time =  21.042 s.
131:
132:  ------------------------------------------------------------------------------
133:   batch[   9 ]  : ncas =      90  : rijk    = 108355911332313.
134:         datalo neutron =    6441  : ncall/s =       1123705.00
135:               cpu time =  27.250 s.
136:
137:  ------------------------------------------------------------------------------
138:   batch[  10 ]  : ncas =     100  : rijk    =  51329540794721.
139:         datalo neutron =    6043  : ncall/s =       1043949.19
140:               cpu time =  26.055 s.
141:
142:  ==============================================================================
143:
```

10.2.4 Summary of Analyz

Next part is the summary of the end of job. The first part of the summary is the number of calls of the analysis subroutine **analyz**.

List 10.10 ● Standard Output List (5)

```
142: ================================================================================
143:
144:  Summary for the end of job
145:
146: --------------------------------------------------------------------------------
147:  number of analyz call vs ncol
148: --------------------------------------------------------------------------------
149:       ncol   number
150:         -4        1   : end of calculation
151:         -3       10   : end of a batch
152:         -2        0   : (unused)
153:         -1        1   : start of calculation
154:          0      196   : fission
155:          1      100   : source
156:          2    95068   : nuclear reaction
157:          3    12411   : termination by energy cut-off
158:          4    14876   : termination by escape or leakage
159:          5        0   : (unused)
160:          6    40795   : absorption reaction
161:          7    68346   : geometry boundary crossing
162:          8        0   : geometry error
163:
```

10.2.5 Summary of Collisions

Next part is the summary of the collisions. The total number of collisions in a region-wise and in a medium-wise is shown. The collision numbers are listed in the Hydrogen collision, the decay reaction, the elastic collisions and non-elastic collisions, respectively.

List 10.11 ● Standard Output List (6)

```
164: --------------------------------------------------------------------------------
165:  Region-wise total number of collisions
166: --------------------------------------------------------------------------------
167:       Region     Hydrogen        Decay      Elastic      Non-Elastic
168:         1             0         1942        24168        36772
169:         2             0          101        17411        27790
170:         3             0           34         9504        18314
171:         4             0           11            5            7
172:
173: --------------------------------------------------------------------------------
174:  Medium-wise total number of collisions
175: --------------------------------------------------------------------------------
176:       Medium     Hydrogen        Decay      Elastic      Non-Elastic
177:         1             0         1942        24168        36772
178:         2             0          101        17411        27790
179:         3             0           34         9504        18314
180:         4             0           11            5            7
181:
```

10.2.6 Summary of Transport Particles

Next part is the summary of the transport particles. The name of particle, kf-code, mass, charge and baryon number are shown.

List 10.12 ● Standard Output List (7)

```
182: ------------------------------------------------------------------------------
183:  List of transport particles
184: ------------------------------------------------------------------------------
185:       Name      kf-code        mass       charge   baryon
186:       proton      2212         938.3        1       1
187:       neutron     2112         939.6        0       1
188:       pion+        211         139.6        1       0
189:       pion0        111         135.0        0       0
190:       pion-       -211         139.6       -1       0
191:       muon+        -13         105.7        1       0
192:       muon-         13         105.7       -1       0
193:       kaon+        321         493.6        1       0
194:       kaon0        311         497.7        0       0
195:       kaon-       -321         493.6       -1       0
196:       gamma         22           0.0        0       0
197:
198:       nu_mu         14           0.0        0       0
199:       eta          221         547.5        0       0
200:       eta'         331         957.8        0       0
201:       Kbar0       -311         497.7        0       0
202:       nbar0      -2112         939.6        0      -1
203:       pbar-      -2212         938.3       -1      -1
204:       Lambda0     3122        1115.7        0       1
205:       Sigma+      3222        1189.4        1       1
206:       Sigma0      3212        1192.5        0       1
207:       Sigma-      3112        1197.4       -1       1
208:       Xi-         3312        1321.3       -1       1
209:
```

10.2.7 Summary of Produced Particles

Next part is the summary of the produced particles.

List 10.13 ● Standard Output List (8)

```
210: ------------------------------------------------------------------------------
211:  prod. particles        number      weight        weight per source
212: ------------------------------------------------------------------------------
213:       neutron           199545    1.9954500E+05     1.9954500E+03
214:       nuleus            134406    1.3440600E+05     1.3440600E+03
215:       proton             13725    1.3725000E+04     1.3725000E+02
216:       gamma               3656    3.6560000E+03     3.6560000E+01
217:       4He                 2781    2.7810000E+03     2.7810000E+01
218:       deuteron            2733    2.7330000E+03     2.7330000E+01
219:       pion-               1789    1.7890000E+03     1.7890000E+01
220:       pion0               1779    1.7790000E+03     1.7790000E+01
221:       pion+               1371    1.3710000E+03     1.3710000E+01
222:       triton              1276    1.2760000E+03     1.2760000E+01
223:       other                305    3.0500000E+02     3.0500000E+00
224:       3He                  244    2.4400000E+02     2.4400000E+00
225:       fission              196    1.9600000E+02     1.9600000E+00
226:       kaon+                114    1.1400000E+02     1.1400000E+00
```

```
 227:       kaon0                83     8.3000000E+01     8.3000000E-01
 228:       kaon-                23     2.3000000E+01     2.3000000E-01
 229:       muon-                 5     5.0000000E+00     5.0000000E-02
 230:       muon+                 4     4.0000000E+00     4.0000000E-02
 231:
 232:  ------------------------------------------------------------------------------
 233:   the other kind of produced particles
 234:  ------------------------------------------------------------------------------
 235:       Name       kf-code        number      mass        charge    baryon
 236:       Lambda0     3122            110     1115.7         0        1
 237:       eta          221            101      547.5         0        0
 238:       Kbar0       -311             25      497.7         0        0
 239:       Sigma-      3112             22     1197.4        -1        1
 240:       Sigma0      3212             13     1192.5         0        1
 241:       eta'         331             11      957.8         0        0
 242:       Sigma+      3222             10     1189.4         1        1
 243:       nu_mu         14              9        0.0         0        0
 244:       Xi-         3312              2     1321.3        -1        1
 245:       pbar-      -2212              1      938.3        -1       -1
 246:       nbar0      -2112              1      939.6         0       -1
 247:
```

10.2.8 Summary of Particles into JAM

Next part is the summary of the particles entered in JAM part.

List 10.14 ● Standard Output List (9)

```
 248:  ------------------------------------------------------------------------------
 249:   the other kind of particles into JAM
 250:  ------------------------------------------------------------------------------
 251:       Name       kf-code        number      mass        charge    baryon
 252:       Lambda0     3122             70     1115.7         0        1
 253:       Sigma-      3112             16     1197.4        -1        1
 254:       Kbar0       -311             15      497.7         0        0
 255:       Sigma+      3222              3     1189.4         1        1
 256:       Xi-         3312              2     1321.3        -1        1
 257:       nbar0      -2112              1      939.6         0       -1
 258:       pbar-      -2212              1      938.3        -1       -1
 259:
```

10.2.9 Summary of Decay Particles

Next part is the summary of the decay particles.

List 10.15 ● Standard Output List (10)

```
 260:  ------------------------------------------------------------------------------
 261:   particle decays      number      weight        weight per source
 262:  ------------------------------------------------------------------------------
 263:       pion0             1779     1.7790000E+03     1.7790000E+01
 264:       other              234     2.3400000E+02     2.3400000E+00
 265:       kaon0               63     6.3000000E+01     6.3000000E-01
 266:       pion-                5     5.0000000E+00     5.0000000E-02
 267:       pion+                3     3.0000000E+00     3.0000000E-02
 268:       kaon+                3     3.0000000E+00     3.0000000E-02
 269:       kaon-                1     1.0000000E+00     1.0000000E-02
```

```
270:
271:  ------------------------------------------------------------------------------
272:   the other kind of decay particles
273:  ------------------------------------------------------------------------------
274:       Lambda0 ->  p+      +  pi-                          41
275:       eta     ->  pi0     +  pi0     +  pi0               36
276:       eta     ->  gamma   +  gamma                        30
277:       eta     ->  pi+     +  pi0     +  pi-               30
278:       Lambda0 ->  n0      +  pi0                          27
279:       Sigma-  ->  n0      +  pi-                          14
280:       Sigma0  ->  Lambda0 +  gamma                        13
281:       Kbar0   ->  pi+     +  pi-                          10
282:       eta'    ->  pi+     +  pi-     +  eta                7
283:       Kbar0   ->  pi0     +  pi0                           7
284:       Sigma+  ->  p+      +  pi0                           5
285:       eta     ->  pi+     +  pi-     +  gamma              5
286:       Sigma+  ->  n0      +  pi+                           4
287:       eta'    ->  pi+     +  pi-                           4
288:       Xi-     ->  Lambda0 +  pi-                           1
289:
```

10.2.10 Summary of Stopped Particles

Next part is the summary of the stopped particles.

List 10.16 ● Standard Output List (11)

```
290:  ------------------------------------------------------------------------------
291:   stop. particles        number      weight        weight per source
292:  ------------------------------------------------------------------------------
293:       nuleus             134406     1.3440600E+05     1.3440600E+03
294:       neutron             55497     5.5497000E+04     5.5497000E+02
295:       proton              11843     1.1843000E+04     1.1843000E+02
296:       gamma                3656     3.6560000E+03     3.6560000E+01
297:       4He                  2781     2.7810000E+03     2.7810000E+01
298:       deuteron             2733     2.7330000E+03     2.7330000E+01
299:       triton               1276     1.2760000E+03     1.2760000E+01
300:       pion+                 508     5.0800000E+02     5.0800000E+00
301:       3He                   244     2.4400000E+02     2.4400000E+00
302:       kaon+                  37     3.7000000E+01     3.7000000E-01
303:       pion-                  12     1.2000000E+01     1.2000000E-01
304:       other                   7     7.0000000E+00     7.0000000E-02
305:       muon-                   3     3.0000000E+00     3.0000000E-02
306:       muon+                   1     1.0000000E+00     1.0000000E-02
307:
308:  ------------------------------------------------------------------------------
309:   the other kind of stopped particles
310:  ------------------------------------------------------------------------------
311:       Name      kf-code        number     mass       charge    baryon
312:       Sigma-       3112             6    1197.4        -1        1
313:       Sigma+       3222             1    1189.4         1        1
314:
```

10.2.11 Summary of Leakage Particles

Next part is the summary of the leakage particles.

List 10.17 ● Standard Output List (12)

```
315: ------------------------------------------------------------------------------
316:  leak. particles        number      weight        weight per source
317: ------------------------------------------------------------------------------
318:      neutron            14322     1.4322000E+04     1.4322000E+02
319:      proton               197     1.9700000E+02     1.9700000E+00
320:      pion-                157     1.5700000E+02     1.5700000E+00
321:      pion+                150     1.5000000E+02     1.5000000E+00
322:      kaon+                 22     2.2000000E+01     2.2000000E-01
323:      other                 16     1.6000000E+01     1.6000000E-01
324:      kaon-                  4     4.0000000E+00     4.0000000E-02
325:      muon+                  3     3.0000000E+00     3.0000000E-02
326:      kaon0                  3     3.0000000E+00     3.0000000E-02
327:      muon-                  2     2.0000000E+00     2.0000000E-02
328:
329: ------------------------------------------------------------------------------
330:  the other kind of leakage particles
331: ------------------------------------------------------------------------------
332:      Name      kf-code        number     mass       charge   baryon
333:      nu_mu          14             9      0.0          0       0
334:      Lambda0      3122             7   1115.7          0       1
335:
```

10.2.12 Summary of Cutoff Particles, Source and CG errors

Next part is the summary of the cutoff particles, neutron and photons, the source normalization, CG errors, and the next initial random number.

List 10.18 ● Standard Output List (13)

```
336: ------------------------------------------------------------------------------
337:  cut-off neutron        number      weight        number per source
338: ------------------------------------------------------------------------------
339:      neutron            55497     5.5497000E+04     5.5497000E+02
340:
341: ------------------------------------------------------------------------------
342:  cut-off gamma          number      weight        number per source
343: ------------------------------------------------------------------------------
344:      gamma               3656     3.6560000E+03     3.6560000E+01
345:
346: ------------------------------------------------------------------------------
347:  source:  maxcas      maxbch      irskip      source normalization
348: ------------------------------------------------------------------------------
349:               10          10           0      100.0000000000000
350:
351: ------------------------------------------------------------------------------
352:  CG error summary
353: ------------------------------------------------------------------------------
354:  Number of lost part in CG    =     0 / nlost = 10
355:  Number of CG region errors   =     0
356:
357: ------------------------------------------------------------------------------
358:  next initial random number =  28487959622441.0
359:
```

10.2.13 Summary of CPU time

The last part is the summary of the cpu time of this job.

List 10.19 ● Standard Output List (14)

```
360:  ------------------------------------------------------------------------------
361:   job termination date : 2000/12/21
362:                   time :   15:33:29
363:
364:  ------------------------------------------------------------------------------
365:      CPU Summary
366:  ------------------------------------------------------------------------------
367:
368:                          sec        %           count
369:  total cpu time =      237.61     100.00
370:
371:       transport =       23.79      10.01
372:        analysis =        2.25       0.95      231693
373:           nevap =       42.53      17.90      133971
374:           nreac =      169.04      71.14      136893
375:
376:           other =        3.37       1.42
377:           hydro =        0.00       0.00           0
378:           dklos =        0.04       0.02        2088
379:           elast =       43.11      18.15       51088
380:           ncasc =      122.51      51.56      104090
381:
382:           berti =       49.75      20.94      103196
383:           isoba =        0.00       0.00           0
384:           JAM   =       70.69      29.75         894
385:
386:     === incident particle into bert =====
387:
388:          proton =                   1.15        1188
389:         neutron =                  94.74       97765
390:            pi + =                   0.83         854
391:            pi - =                   3.28        3389
392:        ----------------------------------------------
393:            total                              103196
394:             real                               82289
395:
396:     === incident particle into JAM =====
397:
398:          proton =                  23.60         211
399:         neutron =                   8.50          76
400:            pi + =                  20.02         179
401:            pi - =                  16.55         148
402:            ka + =                  10.96          98
403:            ka 0 =                   4.14          37
404:            ka - =                   4.14          37
405:           other =                  12.08         108
406:        ----------------------------------------------
407:            total                                 894
408:             real                                 594
409:
410:   END
```

10.3. Output of Track Length Tally

As an example of the track length tally, we take the tally in a calculation by the above input file. We slightly change the input parameters in List 10.1, number of event in a batch and the incident energy as

```
 4:  [ P a r a m e t e r s ]
 5:   maxcas  = 500            <---------
 6:   maxbch  = 10

11:  [ S o u r c e ]
12:   s-type =   1
13:     proj =   proton
14:       e0 =   3000.0        <---------
```

We add the track length tally section and the volume section to the input file.

List 10.20 ● Track Length Tally Section

```
 1:  [ T - T r a c k ]
 2:       mesh = reg
 3:        reg = 1 2 3 4
 4:     e-type = 3
 5:         ne = 31
 6:       emin = 0.8
 7:       emax = 4000.0
 8:       unit = 3
 9:       axis = eng
10:       file = tr-reg.dat
11:       part = proton  neutron  pion+
12:      angel = ymax(5.e-2) ymin(1.e-10) xmin(1) xmax(6.e3) nosp
```

List 10.21 ● Volume Section

```
 1:  [ V o l u m e ]
 2:    reg    vol
 3:     1      4712.4
 4:     2     12959.1
 5:     3     19635.0
 6:     4     50658.2
```

The results of the flux of proton, neutron and π^+ by the track length tally are printed in the file `tr-reg.dat`, which is defined by the line 10 of List 10.20.

The first part of the output file shows the input echo with some comments, from line 1 to 26. This part is also printed in the standard output file.

The next part is the results of flux in the region 1 with statistical errors for proton, neutron and π^+. This continues up to the final region 4.

List 10.22 ● Output of Track Length Tally

```
[ T - T r a c k ]
    title = [t-track] in region mesh
     mesh =  reg          # mesh type is region-wise
      reg = 1 2 3 4
   e-type =    3          # e-mesh is log given by emin, emax and ne
     emin =  0.80000      # minimum value of e-mesh points
     emax =   4000.0      # maximum value of e-mesh points
#    edel =  0.27475      # mesh width of e-mesh points
       ne =   31          # number of e-mesh points
#    data = ( e(i), i = 1, ne + 1 )
#          8.00000E-01  1.05296E+00  1.38590E+00  1.82413E+00  2.40091E+00
#          3.16008E+00  4.15929E+00  5.47446E+00  7.20548E+00  9.48385E+00
#          1.24826E+01  1.64296E+01  2.16247E+01  2.84624E+01  3.74621E+01
#          4.93076E+01  6.48987E+01  8.54196E+01  1.12429E+02  1.47979E+02
#          1.94770E+02  2.56356E+02  3.37416E+02  4.44106E+02  5.84533E+02
#          7.69361E+02  1.01263E+03  1.33283E+03  1.75426E+03  2.30896E+03
#          3.03905E+03  4.00000E+03
     unit =    3          # unit is [1/cm^2/Lethargy/source]
     axis =  eng          # axis of output
     file = tr-reg.dat    # file name of output for the above axis
     part =  proton  neutron    pion+
# kf/name :    2212     2112      211
    angel = ymax(5.e-2) ymin(1.e-10) xmin(1) xmax(6.e3) nosp
#    used :        main (  %)        temp (  %)       total (  %)
#  memory :         778 (  0)          31 (  0)         809 (  0)

#------------------------------------------------------------------------------
#newpage:
#   no. =    1   reg =  1

x: Energy [MeV]
y: Flux [1/cm^2/Lethargy/source]
p: xlog ylog afac(0.8) form(0.9)
p: ymax(5.e-2) ymin(1.e-10) xmin(1) xmax(6.e3) nosp
h: n            x            y(proton ),hh0l  n   y(neutron),hh0dr n   y(pion+  ),hh0mb n
#  e-lower      e-upper         flux     r.err      flux     r.err      flux     r.err
   8.0000E-01   1.0530E+00   5.4460E-07  0.0000   4.9124E-03  0.0000   0.0000E+00  0.0000
   1.0530E+00   1.3859E+00   3.5394E-06  0.0000   2.6651E-02  0.0000   0.0000E+00  0.0000
   1.3859E+00   1.8241E+00   5.0572E-06  0.0000   2.5535E-02  0.0000   6.8106E-11  0.9999
   1.8241E+00   2.4009E+00   7.4467E-06  0.0000   2.3482E-02  0.0000   4.4004E-10  0.9999

      ..................................................
      ..................................................

   1.7543E+03   2.3090E+03   8.1530E-04  0.0421   3.9154E-04  0.0707   7.7983E-06  0.6385
   2.3090E+03   3.0391E+03   1.1172E-02  0.0032   2.7291E-04  0.0878   5.5233E-07  0.9999
   3.0391E+03   4.0000E+03   0.0000E+00  0.0000   0.0000E+00  0.0000   0.0000E+00  0.0000

#   sum over                5.3544E-03  0.0038   6.2321E-02  0.0005   3.4547E-04  0.0242

'no. =    1,    reg  =    1'
msuc: {\huge [t-track] in region mesh}
wt: s(0.7)
\vspace{-3}
   vol  &=&   4.7124E+03 [cm^3]
e:

#------------------------------------------------------------------------------
 newpage:
#   no. =    2   reg =  2

x: Energy [MeV]
y: Flux [1/cm^2/Lethargy/source]
p: xlog ylog afac(0.8) form(0.9)
p: ymax(5.e-2) ymin(1.e-10) xmin(1) xmax(6.e3) nosp
h: n            x            y(proton ),hh0l  n   y(neutron),hh0dr n   y(pion+  ),hh0mb n
#  e-lower      e-upper         flux     r.err      flux     r.err      flux     r.err
   8.0000E-01   1.0530E+00   2.4900E-08  0.0000   1.5148E-03  0.0019   0.0000E+00  0.0000
   1.0530E+00   1.3859E+00   1.6134E-07  0.0000   8.2780E-03  0.0000   0.0000E+00  0.0000
   1.3859E+00   1.8241E+00   2.2980E-07  0.0000   7.7529E-03  0.0000   0.0000E+00  0.0000
```

```
 91:     1.8241E+00   2.4009E+00   3.3772E-07  0.0000   6.7987E-03  0.0000   0.0000E+00  0.0000

         ..................................................
         ..................................................

218:     1.7543E+03   2.3090E+03   7.1562E-06  0.0995   1.8667E-06  0.2118   0.0000E+00  0.0000
219:     2.3090E+03   3.0391E+03   3.5921E-04  0.0001   3.6086E-07  0.4470   0.0000E+00  0.0000
220:     3.0391E+03   4.0000E+03   0.0000E+00  0.0000   0.0000E+00  0.0000   0.0000E+00  0.0000
221:
222:  #   sum over                  1.2516E-04  0.0077   3.7412E-03  0.0013   8.4322E-06  0.0664
223:
224:  'no. =    4,    reg  =    4'
225:  msuc: {\huge [t-track] in region mesh}
226:  wt: s(0.7)
227:  \vspace{-3}
228:     vol  &=&   5.0658E+04 [cm^3]
229:  e:
```

If this file is processed by *ANGEL* [58], the EPS(ENHANCED POSTSCRIPT) file is obtained. The EPS file shows the following four figures.

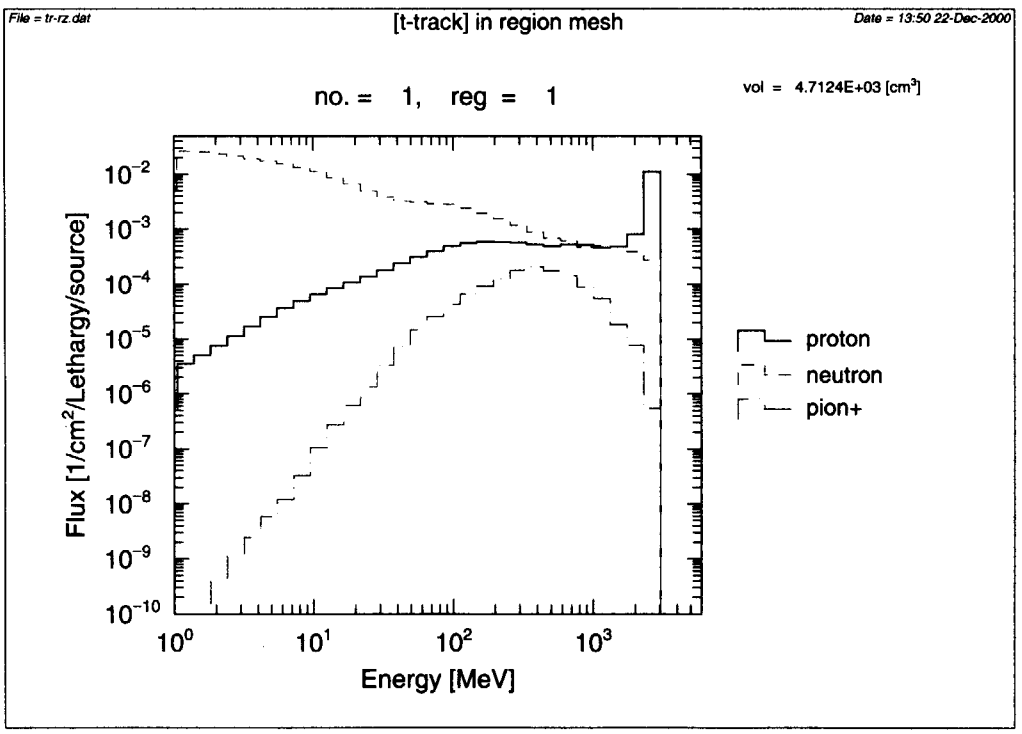

Figure 50. Flux by track length tally, page 1.

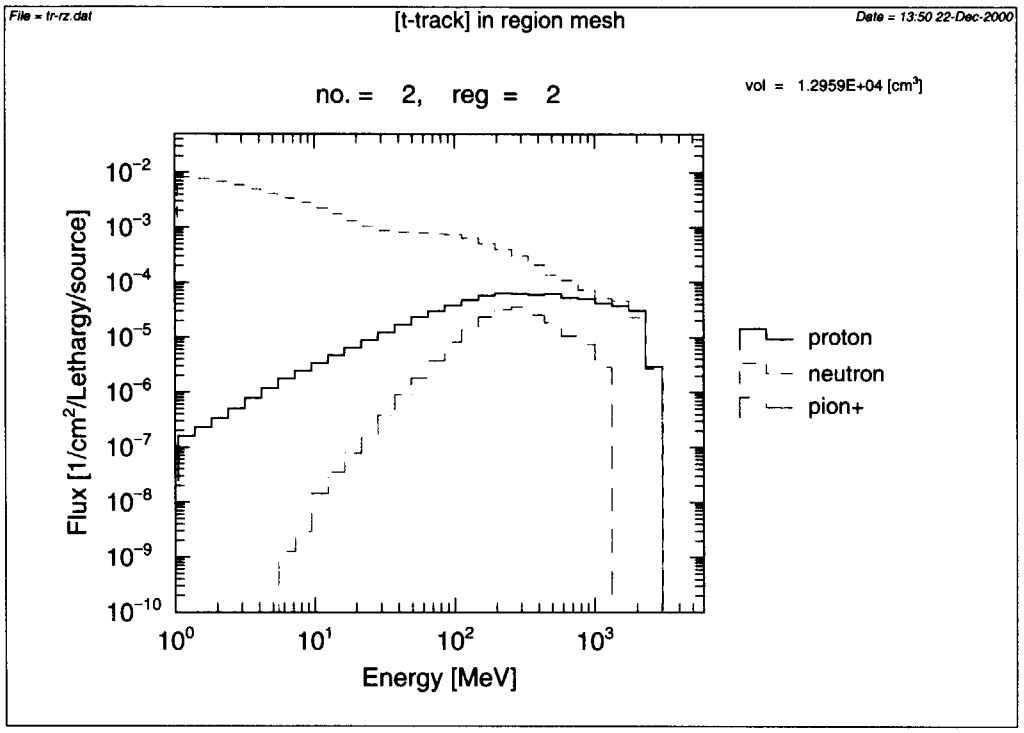

Figure 51. Flux by track length tally, page 2.

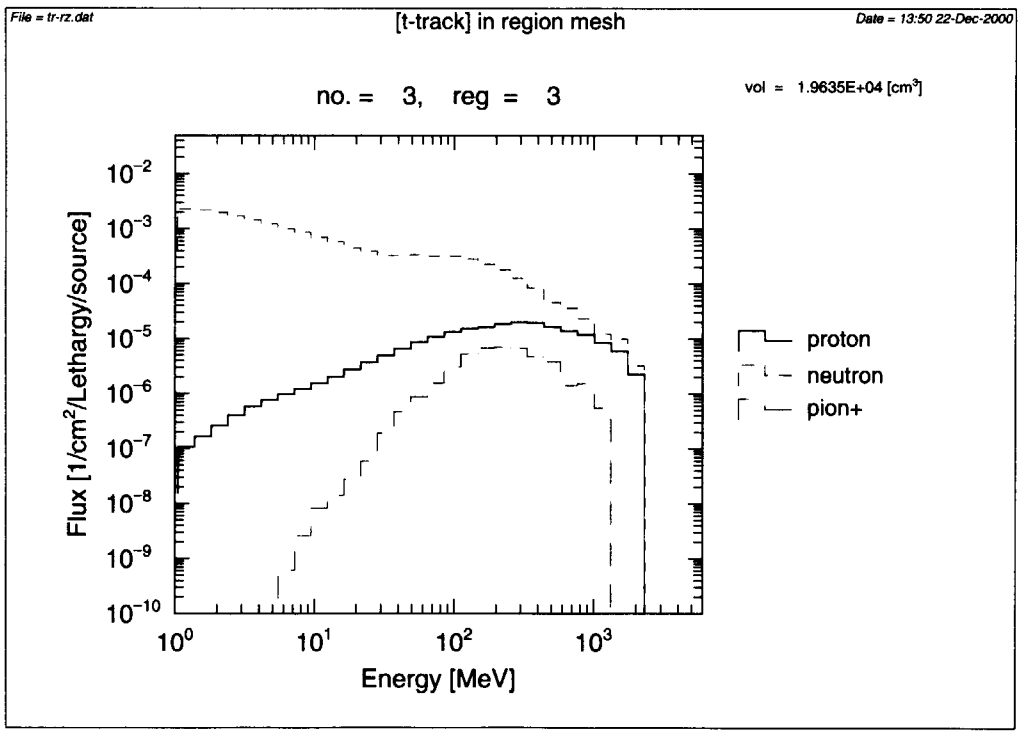

Figure 52. Flux by track length tally, page 3.

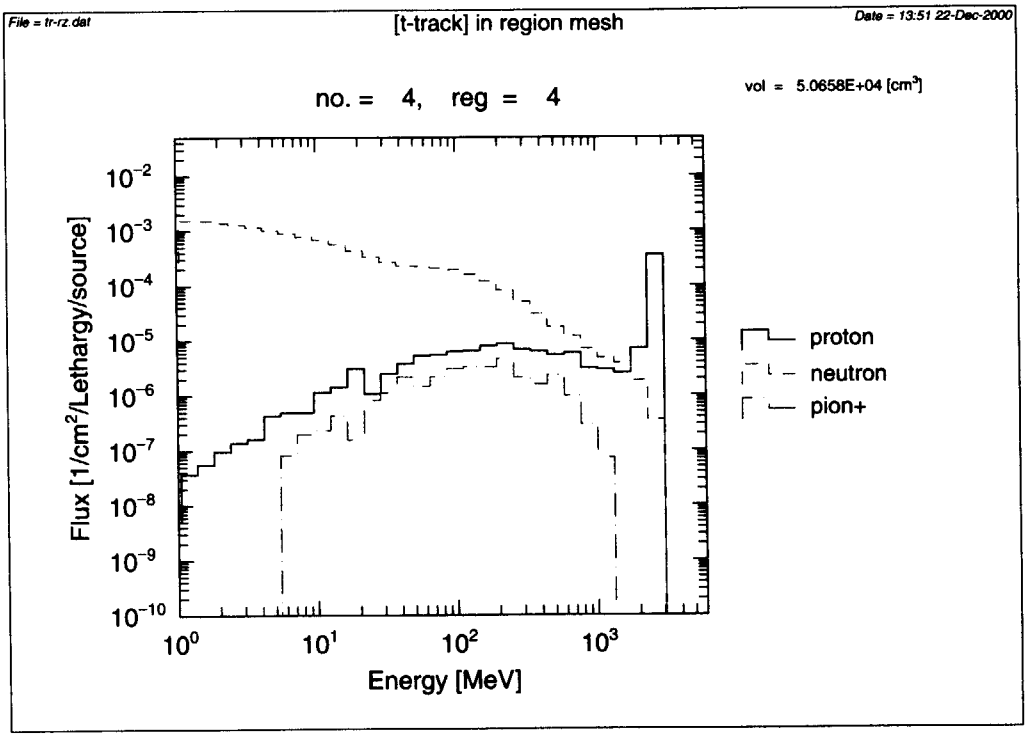

Figure 53. Flux by track length tally, page 4.

10.4. Output of Surface Crossing Tally

With the following input for the surface crossing tally, we calculate the same system as in the previous section.

List 10.23 ● Surface Crossing Tally Section

```
[ T - C r o s s ]
      mesh = xyz
    x-type =  2
       nx =  80
      xmin = -20
      xmax =  20
    y-type =  2
       ny =  80
      ymin = -20
      ymax =  20
    z-type =  1
       nz =  3
       2.5 12.5 22.5 32.5
      unit = 1
      axis = xy
      file = cr-xyz.dat
    output = of-curr
      part = all
   2d-type = 3
```

The results are shown in the following figures after processed by *ANGEL* [58].

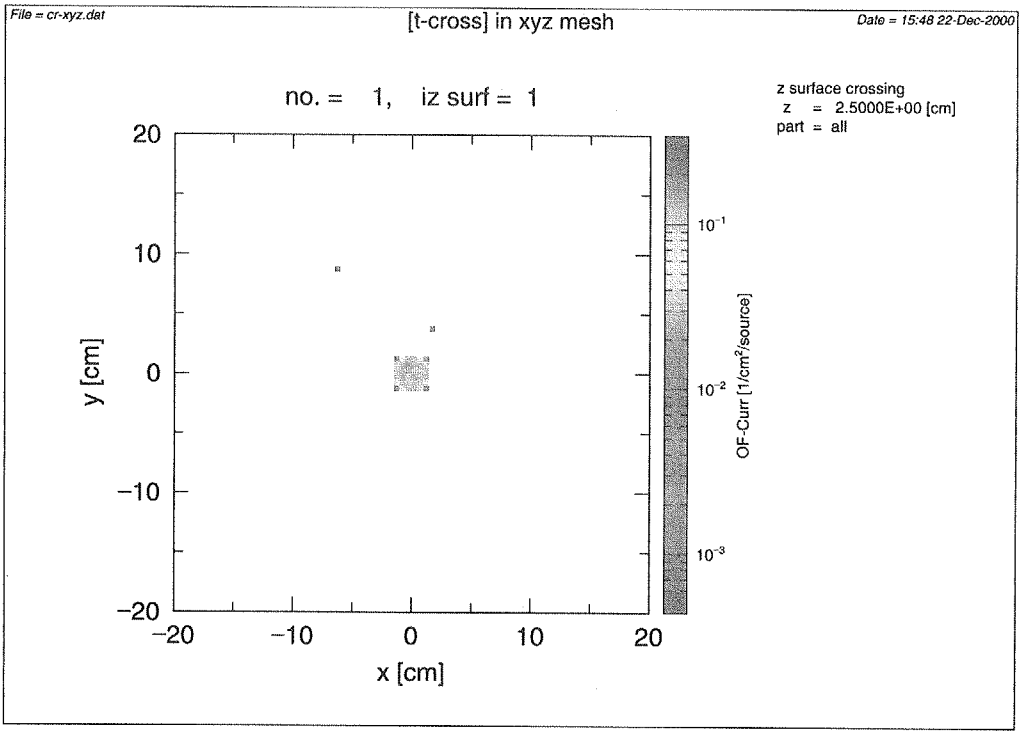

Figure 54. Density of Crossing Particles, page 1.

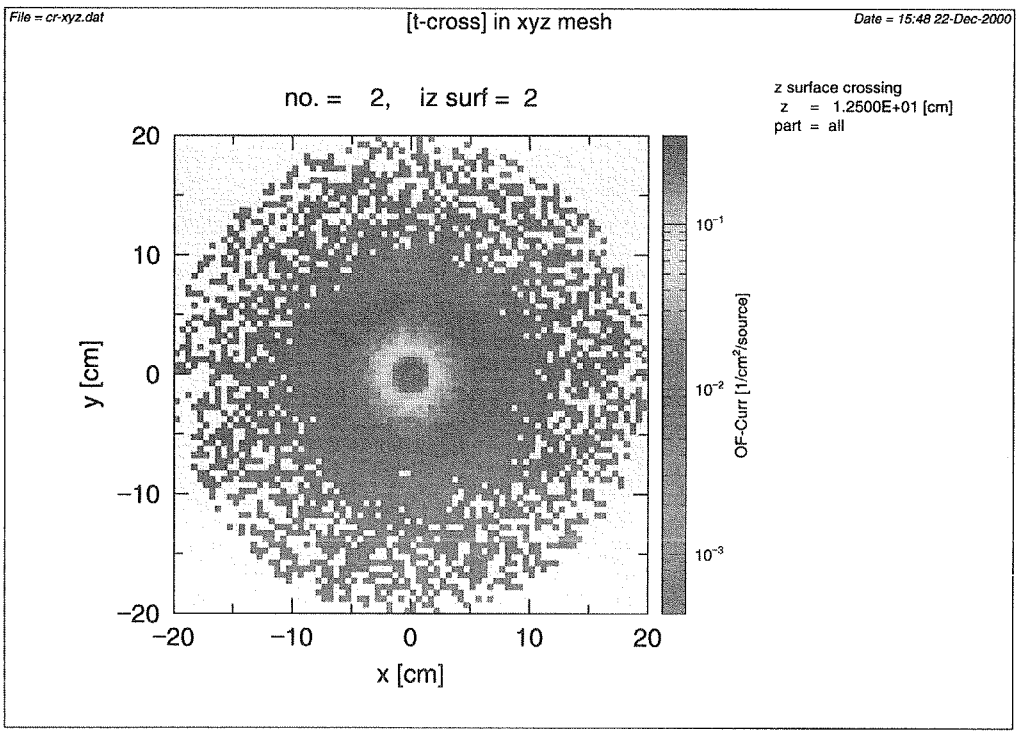

Figure 55. Density of Crossing Particles, page 2.

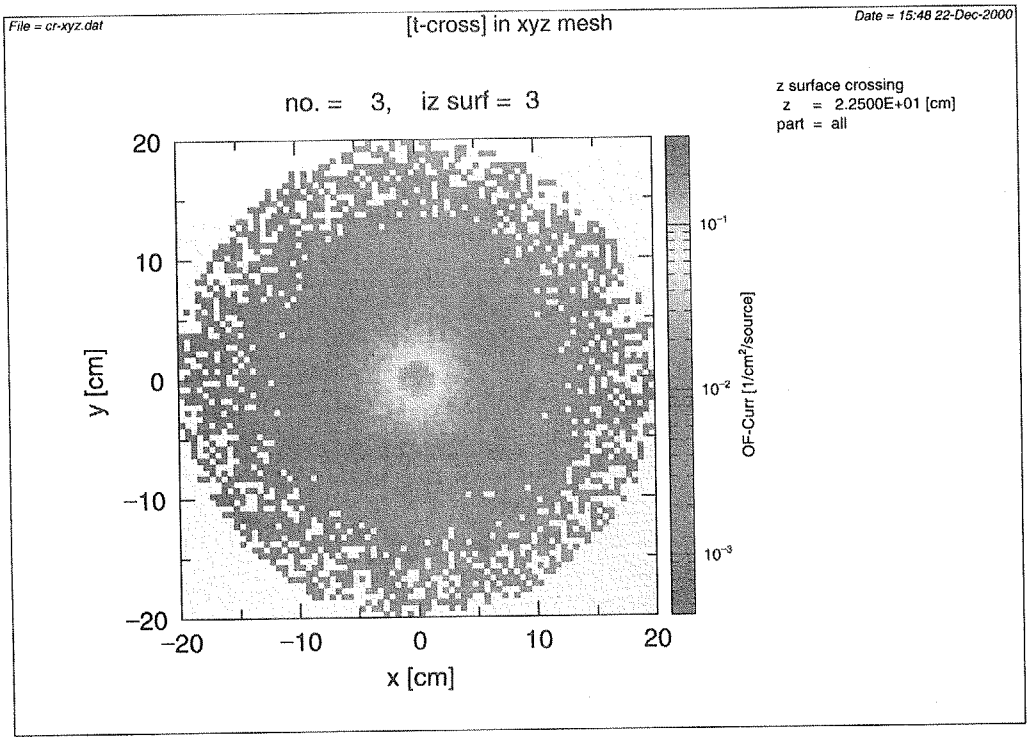

Figure 56. Density of Crossing Particles, page 3.

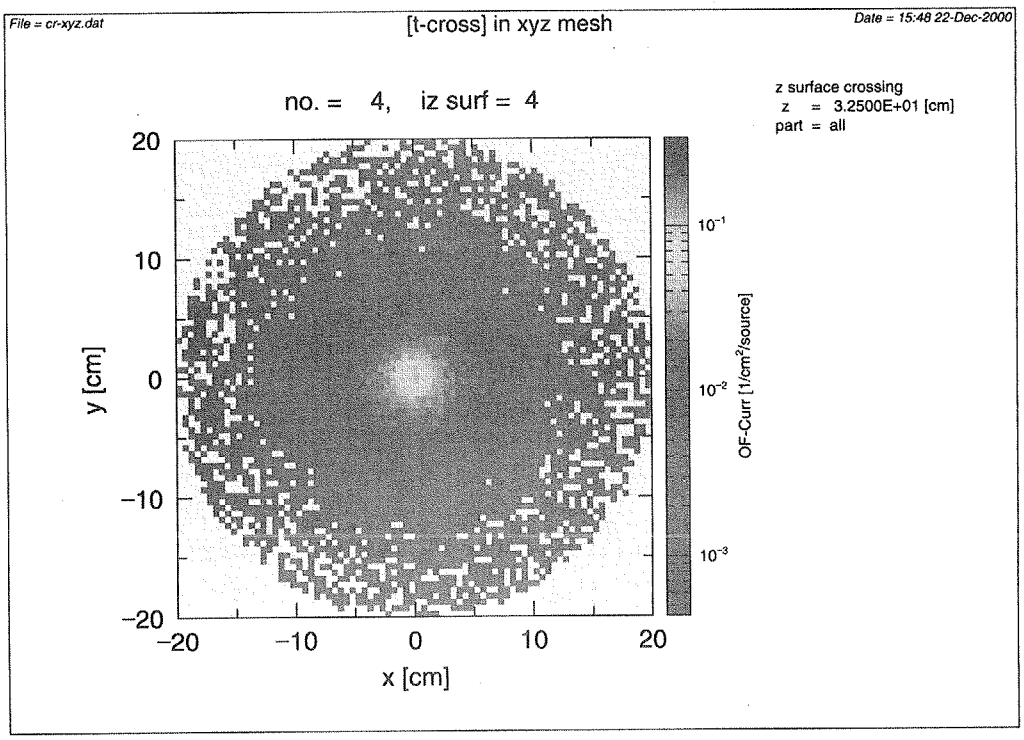

Figure 57. Density of Crossing Particles, page 4.

10.5. Output of Nuclide Yield Tally

With the following input for the nuclide yield tally, we calculate the same system as in the previous section.

List 10.24 ● Nuclide Yield Tally Section

```
[ T - Y i e l d ]
     mesh = r-z
   r-type =  2
       nr =  1
     rmin =  0
     rmax =  20
   z-type =  2
       nz =  1
     zmin =  0
     zmax =  60
     unit = 1
     axis = mass charge chart
     file = y-mass.dat y-charg.dat y-chart.dat
     info = 1
```

The results are shown in the following figures after processed by *ANGEL* [58].

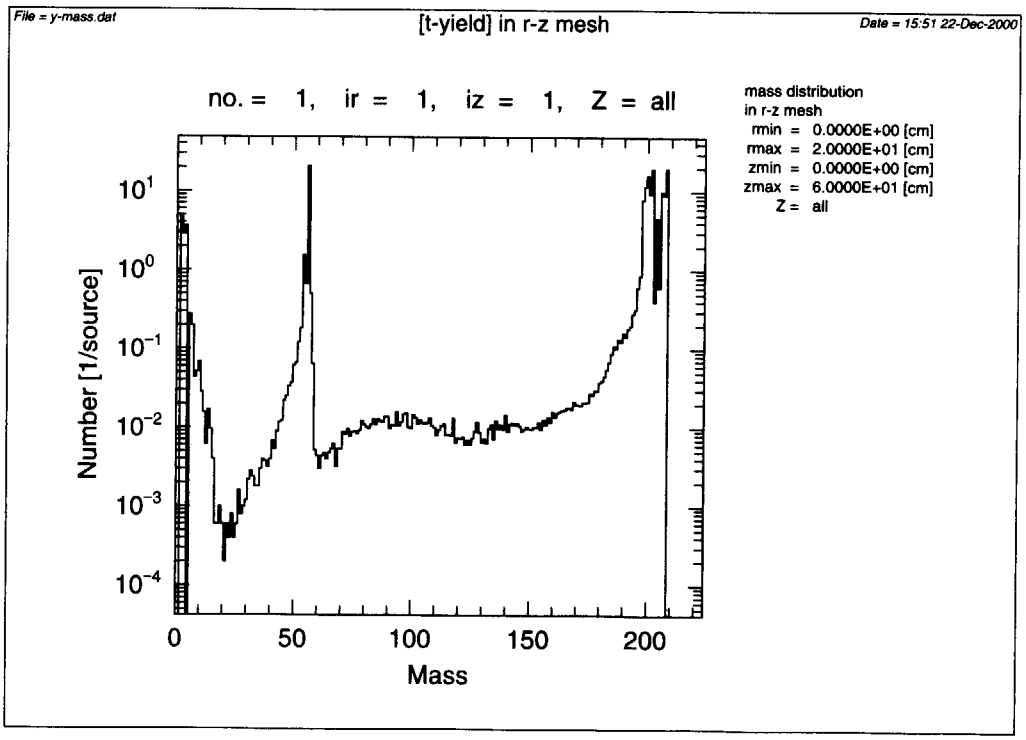

Figure 58. Mass Distribution.

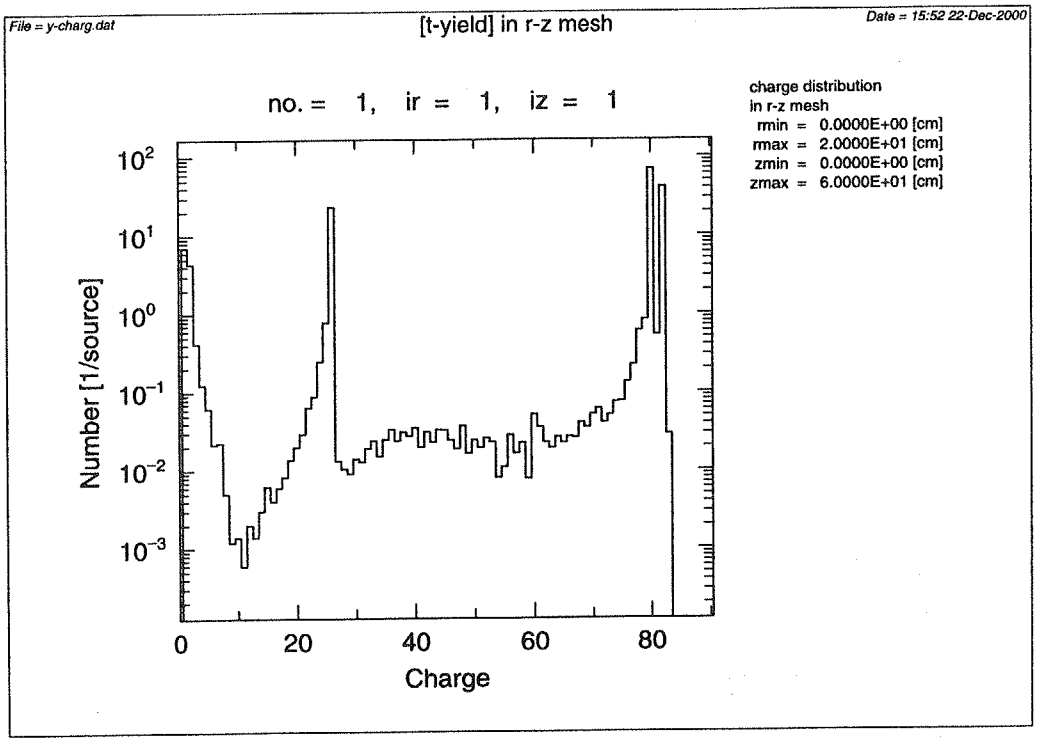

Figure 59. Charge Distribution.

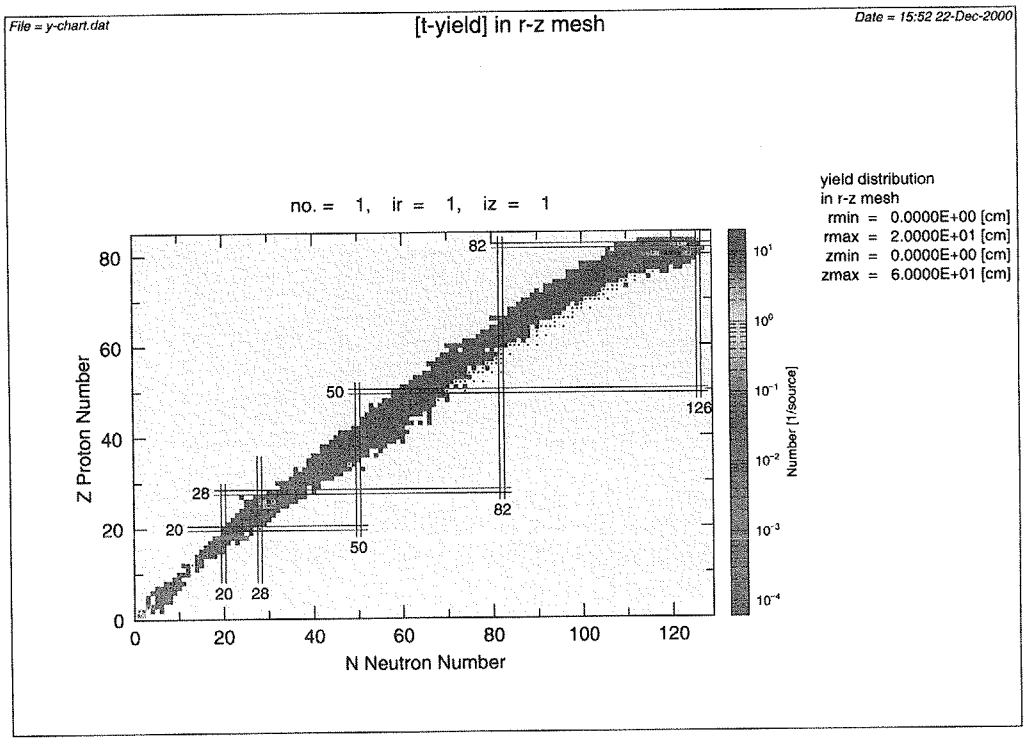

Figure 60. Yield Distribution in Nuclear Chart.

10.6. Output of Heat Tally

With the following input for the heat tally, we calculate the same system as in the previous section.

List 10.25 ● Heat Tally Section

```
 1:  [ T - H e a t ]
 2:       mesh = r-z
 3:     r-type =  2
 4:         nr =  40
 5:       rmin =  0
 6:       rmax =  20
 7:     z-type =  2
 8:         nz =  60
 9:       zmin =  0
10:       zmax =  60
11:       axis = rz
12:       file = h-rz.dat
13:     output = heat
14:       unit = 1
15:      angel = form(0.6666)
```

The results are shown in the following figures after processed by ANG_EL [58].

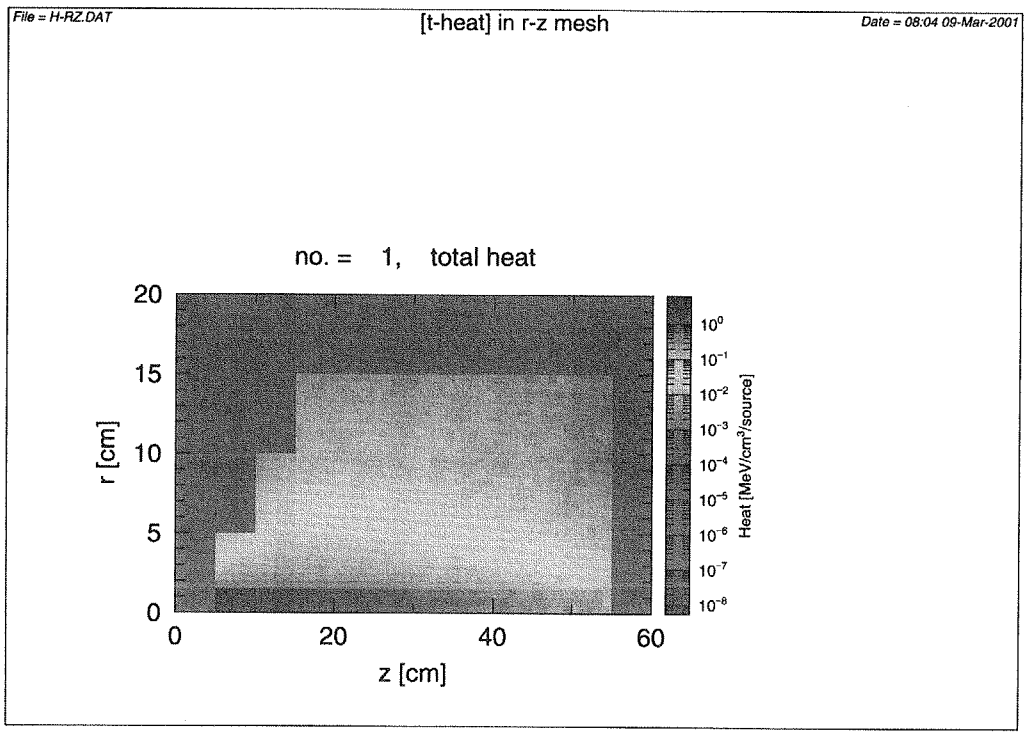

Figure 61. Heat Distribution.

10.7. Output of Star Density Tally

With the following input for the star density tally, we calculate the same system as in the previous section.

List 10.26 ● Star Density Tally Section

```
 1:  [ T - S t a r ]
 2:       mesh = xyz
 3:     x-type =  1
 4:         nx =  1
 5:         -5.0 5.0
 6:     y-type =  2
 7:         ny =  40
 8:       ymin = -20
 9:       ymax =  20
10:     z-type =  2
11:         nz =  60
12:       zmin =  0
13:       zmax =  60
14:     e-type =  2
15:         ne =  1
16:       emin =  0
17:       emax =  4000.0
18:       unit = 1
19:       axis = yz
20:       file = s-yz.dat
```

The results are shown in the following figures after processed by *ANGEL* [58].

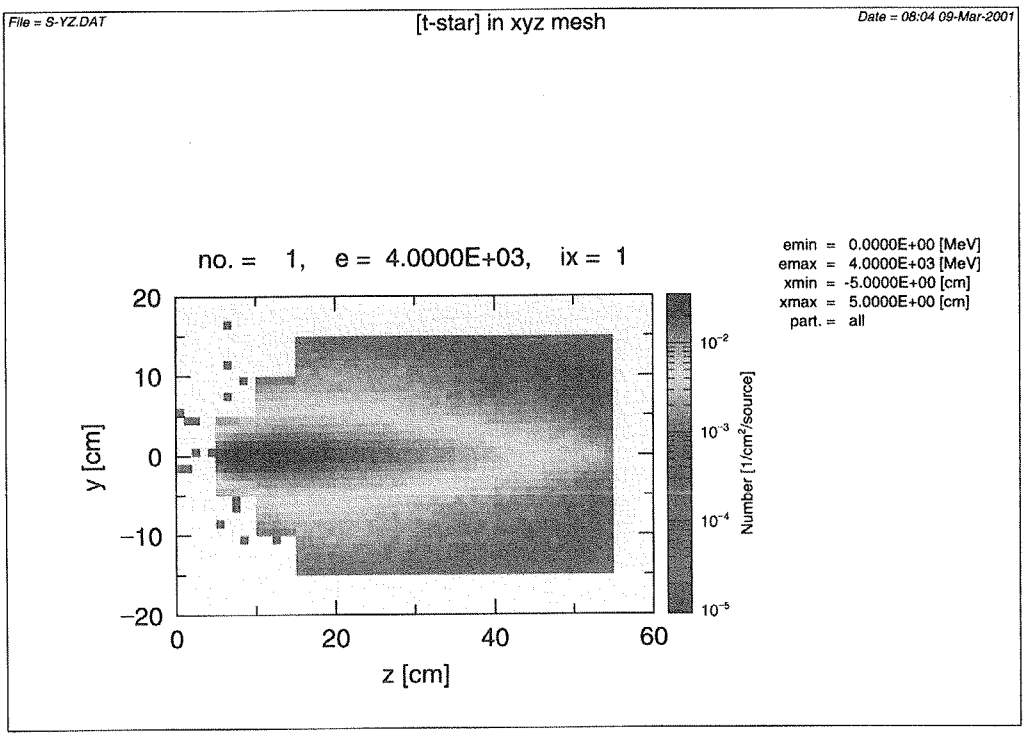

Figure 62. Star Density Distribution.

10.8. Output of Time Tally

With the following input for the time tally, we calculate the same system as in the previous section.

List 10.27 ● Time Tally Section

```
[ T - T i m e ]
     mesh = r-z
   r-type =  2
       nr =  1
     rmin =  0
     rmax =  21
   z-type =  2
       nz =  1
     zmin =  -1
     zmax =  61
   e-type =  2
       ne =  1
     emin =  0
     emax =  4000.0
   t-type =  2
       nt =  40
     tmin =  0
     tmax =  40.0
     unit =  2
     part = neutron
     axis =  t
     file =  t-rz.dat

   output = escape
   output = cutoff
```

The results are shown in the following figures after processed by ANG_EL [58] for escape and cutoff cases.

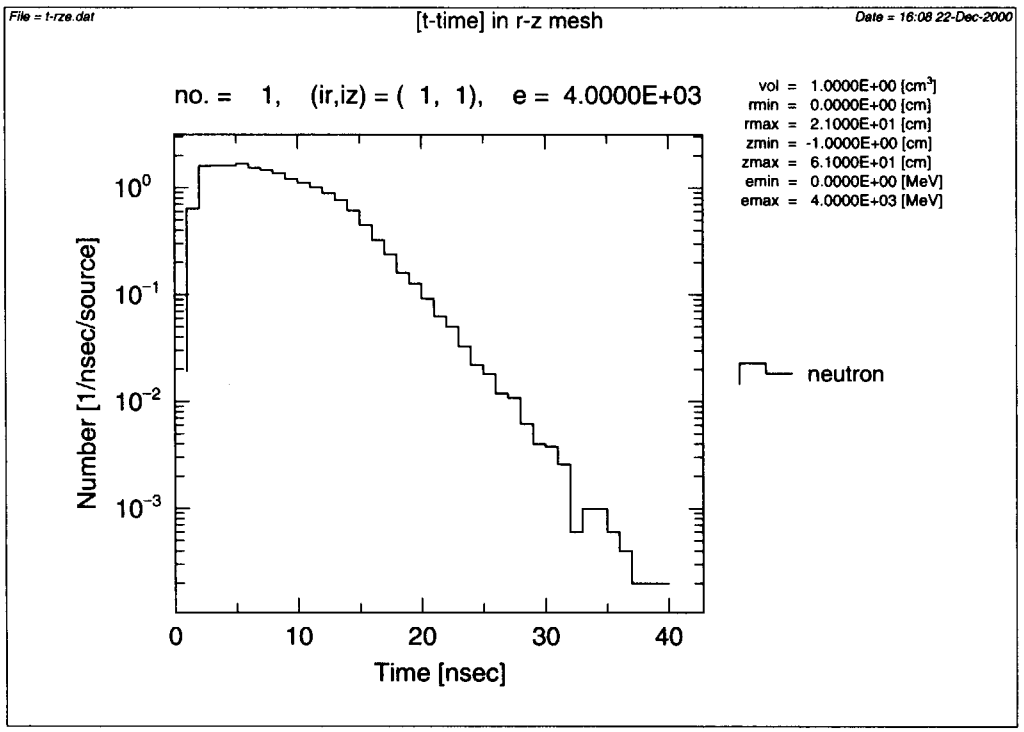

Figure 63. Time Spectrum of Escape Neutron.

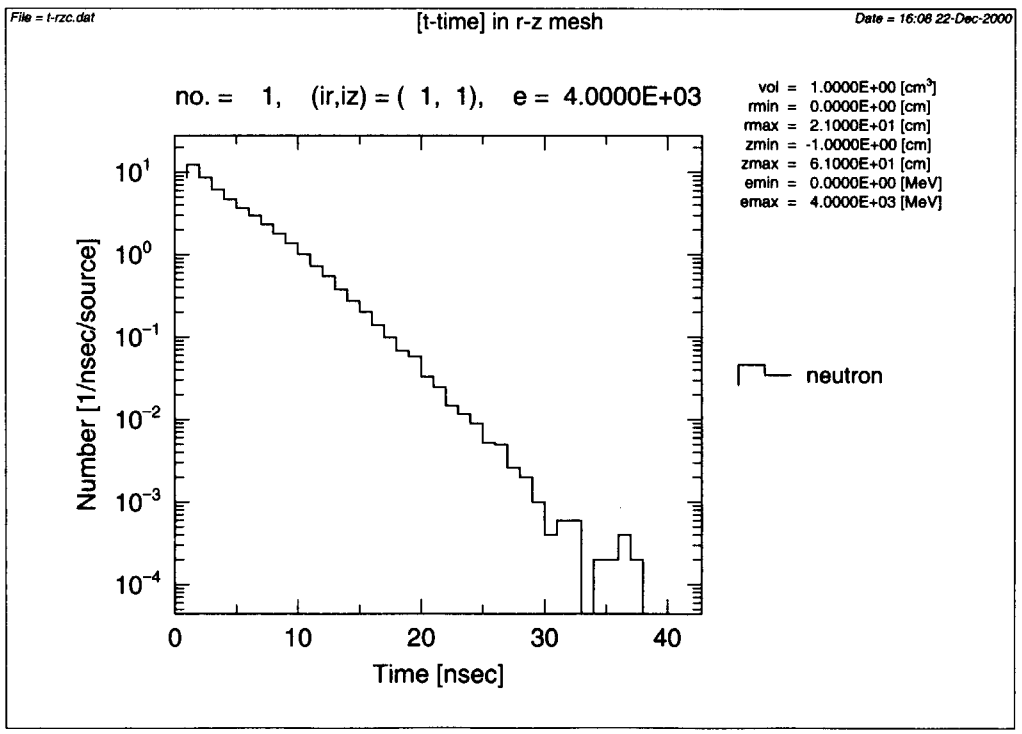

Figure 64. Time Spectrum of Cutoff Neutron.

11. Summary

We have developed a high energy nucleon-meson transport code NMTC/JAM, which is an upgrade version of NMTC/JAERI97. We have introduced two new physical models in NMTC/JAM, the high energy nuclear reaction code JAM for the intra-nuclear cascade part and the generalized evaporation model GEM for the evaporation and fission process. The former can extend the applicable energy range of NMTC/JAM in principle up to 200 GeV for nucleons and mesons. While the latter enable us to calculate the light nucleus production like Be isotopes from the excited residual nucleus. These nuclei are very important in view of the radiation safety for estimating induced radiation activities and dose rate in the spallation neutron source facility driven by a high intensity proton beam.

According to the increase of the applicable energy, we have upgraded the nucleon-nucleus non-elastic, elastic and differential elastic cross section data by employing new systematics based on Pearlstein's systematics.

In addition, the particle transport in a magnetic field has been implemented for the beam transport calculations. By this new function, we can estimate the heat and the activation of the material around the beam, magnets, beam scraper and beam window and we can carry out the design of the facilities concerning the beam transport.

In the present upgrading work, we have rewritten the whole source code except for the old packages of, e.g., Bertini model, DRES package, CG package, since the original source had written in an old Fortran style. By this renewal of the source code, we can easily add new utilities to the code. We have implemented two new tally functions, the star density tally and time tally, and improved very much the format of input and output of data in a user friendly manner. Particularly, the output of the tally functions can be immediately shown on the screen after the calculation.

For the code validation, we have compared the calculations by NMTC/JAM code with the experimental data of thick targets measured at KEK-PS and AGS for proton induced reactions up to several tens GeV. The results of NMTC/JAM code show excellent agreement with the experimental data.

From these new calculation functions and utilities implemented in the code and the code validation, consequently, NMTC/JAM enables us to carry out reliable neutronics study of a large scale target system with complex geometry more accurately and easily than before.

Acknowledgements

The authors would like to thank Dr. S. Furihata of MRI for supplying the GEM code. They also thank Dr. Y. Nara for his useful comments on the JAM code. They are grateful to Dr. T. Fukahori and Dr. S. Chiba for their useful discussions about parametrization of the nucleon-nucleus cross sections.

References

1) Nakahara Y., Tsutsui T.: "NMTC/JAERI A Code System for High Energy Nuclear Reactions and Nucleon-Meson Transport Code", JAERI-M 82-198, (1982) [in Japanese].

2) Coleman W. A., Armstrong T. W.: "The Nucleon-Meson Transport Code NMTC", ORNL-4606, (1970).

3) Bertini H. W., Guthrie M. P., Hermann O. W.: "Instructions for the Operation of Code Associated with MECC-3, A Preliminary Version of an Intranuclear Cascade Calculations for Nuclear Reaction", ORNL-4564, (1971).

4) Dresner L.: "EVAP: A Fortran Program for Calculating the Evaporation of Various Particles from Excited Compound Nuclei", ORNL-TM-190, (1962).

5) Nakahara Y.: J. Nucl. Sci. Technol., **20**, 511 (1983).

6) Takada H., Yoshizawa N., Kosaka K. and Ishibasi K.: "An Upgrade Version of The Nucleon Meson Transport Code : NMTC/JAERI97", JAERI-Data/Code 98-005, Japan Atomic Energy Research Institute (1998).

7) Pearlstein S.: Nucl. Sci. Eng., **49**, 162 (1972); Nucl. Sci. Eng., **95**, 116 (1987); Astrophs. J. **346**, 1049 (1989).

8) West J. T., Emmett M. B.: "MARS: A Multiple Array System Using Combinatorial Geometry", Vol. 3, Section M9 of NUREG/CR-200 (1984).

9) Sato O., Iwai S., Nakamura M., Uehara T., Takagi S., Hirayama H.: "UCMARS: A User Code with a Multiple-Array System using Combinatorial Geometry for EGS4", KEK Internal 94-12, KEK, (1994).

10) The Joint Project Team of JAERI and KEK: "The Joint Project for High-Intensity Proton Accelerator," JAERI-Tech 99-056, Japan Atomic Energy Research Institute (1999).

11) Breismaister J. F. (Ed.): "MCNP A General Monte Carlo N-Particle Transport Code, Version 4A", LA-12625, Los Alamos National Laboratory (1993).

12) Takada H. et al.: "Measurements of Reaction Rate Distributions on a Mercury Target Bombarded with High Energy Protons," *Proc. of the 14th Mtg. of the Int. Collaboration on Advanced Neutron Sources*, Illinois, 1998, Vol.II, ANL-98-33, Argonne National Laboratory (1998).

13) Nara Y., Otuka N., Ohnishi A., Niita K. and Chiba S.: "Study of Relativistic Nuclear Collisions at AGS Energies from p+Be to Au+Au with Hadronic Cascade Model," Phys, Rev. **C61**, 024901 (1999); *http://hadron31.tokai.jaeri.go.jp/jam/.*

14) Furihata S.: Nucl. Instr. and Meth. **B171**, 251, (2000).

15) Lehmer D. H.: "Mathematical Methods in Large-Scale Computing Units," Am. Comput. Lab., Harvard Univ., 26, 141-146 (1951).

16) Particle-Data-Group: Phys. Rev. **D54**, (1996).

17) Wang X. N. and Gyulassy M.: Phys. Rev. **D44**, 3501 (1991); Wang X. N.: Phys. Rep. **280**, 287 (1997); Wang X. N. and Gyulassy M.: Comp. Phys. Comm. **83**, 307 (1994); *http://www-nsdth.lbl.gov/~xnwang/hijing/.*

18) Sjöstrand T.: Comp. Phys. Comm. **82**, 74 (1994); PYTHIA 5.7 and JETSET 7.4 Physics and Manual. *http://thep.lu.se/tf2/staff/torbjorn/Welcome.html.*

19) Zhang B.: Comp. Phys. Comm. **109**, 70 (1997); *http://nt1.phys.columbia.edu/people/bzhang/ZPC/zpc.html*; Zhang B. and Pang Y.: Phys. Rev. **C56**, 2185 (1997); Zhang B., Gyulassy M. and Pang Y.: Phys. Rev. **C58**, 1175 (1998).

20) Kortemeyer G., Bauer W., Haglin K., Murray J. and Pratt S.: Phys. Rev. **C52**, 2714 (1995).

21) Andersson B., Tai A. and Ben-Hao Sa.: Z. Phys. **C70**, 499 (1996).

22) Bonn-Hamburg-München Collaboration, Blobel V., et al.: Nucl. Phys. **B69**, 454 (1974).

23) "Total Cross-Sections for Reactions of High Energy Particles" vol. 12a and vol. 12b edited by Baldni A., Flaminio V., Moorhead W. G. and Morrison D. R. O. (Springer-Verlag Berlin 1988).

24) High-Energy Reactions Analysis Group, CERN Report CERN-HERA 83-01 and 83-02, 1983 (unpublished).

25) E802 Collaboration, Abbott T., et al.: Phys. Rev. **D45** 3906 (1992); Compilation of Relativistic Heavy-Ion Data, *http://www.nndc.bnl.gov/~hi_data/rhid.html.*

26) Ishibashi K., et al.: J. Nucl. Sci. Technol. **32**, 827 (1995)

27) En'yo H., et al.: Phys. Lett. **159B**, 1 (1985).

28) Bertini H. W.: "Monte Carlo Calculations on Intranuclear Cascade", ORNL-3833, Oak Ridge National Laboratory (1963).

29) Cugnon J., Mizutani T., Vandermeulen J.: Nucl. Phys. **A352**, 505 (1981) ; Cugnon J. : Phys. Rev. **C22**, 1885 (1980).

30) Steinheimer R. M., Lindenbaum S. J.: Phys. Rev. **105**, 1874 (1957); ibid., **109**, 1723 (1958) ; ibid., **123**, 333 (1961).

31) Chen K., Fraenkel Z., Friedlander G., Grover J. R., Miller M., Shimamoto Y.: Phys. Rev. **166**, 949 (1968); Harp G. D., Chen K., Friedlander G., Fraenkel Z., Miller M.: Phys. Rev. **C8**, 581 (1973) ; Fraenkel Z.: Private communication.

32) Chen K., Friedlander G., Harp G. D., Miller M.: Phys. Rev. **C4**, 2234 (1971).

33) For Example, Tuma J.J.: Handbook of Physical Calculations, p.251, McGraw-Hill Book Company (1976).

34) Takada H. : J. Nucl. Sci. Technol. **33**, 275 (1996).

35) Niita K., Chiba S., Maruyama Tosh., Maruyama Tomo., Takada H., Fukahori T., Nakahara Y., and Iwamoto A.: Phys. Rev. **C52**, 2620, (1995).

36) Charity R. J. et al.: Nucl. Phys. **A483**, 371, (1988).

37) Audi G. and Wapstra A. H.: Nucl. Phys. **A595**, 409, (1995).

38) Moller P., Nix J. R., Myers W. D., and Swiatecki W. J.: Atomic Data and Nuclear Data Tables, **59**, 185, (1995).

39) Haustein P. E. : Atomic Data and Nuclear Data Tables, **39**, 185, (1988).

40) Gilbert A. and Cameron A. G. W. : Can. J. Phys. **43**, 1446, (1965).

41) Data extracted using the National Nuclear Data Center (NNDC) On-Line Data Service from the ENSDF database, file revised as of August 5 (1998).

42) Green R. E. L., Korteling R. G., and Jackson K. P.: Phys. Rev. **C29**, 1806 (1984).

43) Guthrie M. P.: "EVAP-4 : Another Modification of a Code to Calculate Particle Evaporation from Excited Compound Nuclei", ORNL-TM-3119, (1970).

44) Baba H.: Nucl. Phys. **A159**, 625 (1970).

45) Ignatyuk A. V., Smirenkin G. N., Tishin A. S.: Sov. J. Nucl. Phys. **21**, 256 (1975).

46) Mengoni A., Nakajima Y.: J. Nucl. Sci. Technol. **31**, 151 (1994).

47) Il'inov A. S., Charpanov E. A. and Chirginov S. E.: Sov. J. Nucl. Phys. **32**, 166 (1980).

48) Kupriyanov V. M., et al. : Sov. J. Nucl. Phys. **32**, 183 (1980).

49) Neuzil E. F., Fairhal A. W.: Phys. Rev. **129**, 2705 (1963).

50) Grass R. A., et al.: Phys. Rev. **104**, 404 (1956).

51) Pik-Pichak G. A., Strutinskii V. M.: "Physics of Nuclear Fission", p.8, edited by N. A. Perfilov and V. S. Eismont, Israel Program for Scientific Translation, 1964.

52) Gudima K. K., Mashnik, S. G., Toneev V.D.: Nucl. Phys. **A401**, 329 (1983).

53) Nakahara Y., Nishida T.: "Monte Carlo Algorithms for Simulating Particle Emissions from Pre-equilibrium States during Nuclear Spallation Reactions", JAERI-M 86-074, (1986).

54) Chadwick M. B., et al.: "Cross-Section Evaluation to 150 MeV for Accelarato-Driven Systems and Implementation in MCNPX", Nucl. Sci. Eng. **131**, 293 (1999).

55) "NEA Data Bank Nuclear Data Services: Experimetal Data" *http://www.nea.fr/html/dbdata/nds_experimental.htm.*

56) Ibaraki M., et al.: Nucl. Instr. and Meth. **A446**, 536, (2000).

57) Arai M., et al., J. Neutron Research, vol.8, 71 (1999).

58) Niita K.: "User's Manual, *ANGEL* ver.3.03", unpublished.

国際単位系 (SI) と換算表

表 1　SI 基本単位および補助単位

量	名称	記号
長さ	メートル	m
質量	キログラム	kg
時間	秒	s
電流	アンペア	A
熱力学温度	ケルビン	K
物質量	モル	mol
光度	カンデラ	cd
平面角	ラジアン	rad
立体角	ステラジアン	sr

表 2　SI と併用される単位

名称	記号
分，時，日	min，h，d
度，分，秒	°，′，″
リットル	l，L
トン	t
電子ボルト	eV
原子質量単位	u

1 eV = 1.60218 × 10^{-19} J

1 u = 1.66054 × 10^{-27} kg

表 5　SI 接頭語

倍数	接頭語	記号
10^{18}	エクサ	E
10^{15}	ペタ	P
10^{12}	テラ	T
10^{9}	ギガ	G
10^{6}	メガ	M
10^{3}	キロ	k
10^{2}	ヘクト	h
10^{1}	デカ	da
10^{-1}	デシ	d
10^{-2}	センチ	c
10^{-3}	ミリ	m
10^{-6}	マイクロ	μ
10^{-9}	ナノ	n
10^{-12}	ピコ	p
10^{-15}	フェムト	f
10^{-18}	アト	a

表 3　固有の名称をもつ SI 組立単位

量	名称	記号	他の SI 単位による表現
周波数	ヘルツ	Hz	s^{-1}
力	ニュートン	N	m·kg/s^2
圧力，応力	パスカル	Pa	N/m^2
エネルギー，仕事，熱量	ジュール	J	N·m
工率，放射束	ワット	W	J/s
電気量，電荷	クーロン	C	A·s
電位，電圧，起電力	ボルト	V	W/A
静電容量	ファラド	F	C/V
電気抵抗	オーム	Ω	V/A
コンダクタンス	ジーメンス	S	A/V
磁束	ウェーバ	Wb	V·s
磁束密度	テスラ	T	Wb/m^2
インダクタンス	ヘンリー	H	Wb/A
セルシウス温度	セルシウス度	℃	
光束	ルーメン	lm	cd·sr
照度	ルクス	lx	lm/m^2
放射能	ベクレル	Bq	s^{-1}
吸収線量	グレイ	Gy	J/kg
線量当量	シーベルト	Sv	J/kg

表 4　SI と共に暫定的に維持される単位

名称	記号
オングストローム	Å
バーン	b
バール	bar
ガル	Gal
キュリー	Ci
レントゲン	R
ラド	rad
レム	rem

1 Å = 0.1 nm = 10^{-10} m

1 b = 100 fm^2 = 10^{-28} m^2

1 bar = 0.1 MPa = 10^5 Pa

1 Gal = 1 cm/s^2 = 10^{-2} m/s^2

1 Ci = 3.7 × 10^{10} Bq

1 R = 2.58 × 10^{-4} C/kg

1 rad = 1 cGy = 10^{-2} Gy

1 rem = 1 cSv = 10^{-2} Sv

（注）

1. 表 1 - 5 は「国際単位系」第 5 版，国際度量衡局 1985 年刊行による。ただし，1 eV および 1 u の値は CODATA の 1986 年推奨値によった。
2. 表 4 には海里，ノット，アール，ヘクタールも含まれているが日常の単位なのでここでは省略した。
3. bar は，JIS では流体の圧力を表わす場合に限り表 2 のカテゴリーに分類されている。
4. EC 閣僚理事会指令では bar，barn および「血圧の単位」mmHg を表 2 のカテゴリーに入れている。

換算表

力

N(=10^5 dyn)	kgf	lbf
1	0.101972	0.224809
9.80665	1	2.20462
4.44822	0.453592	1

粘度　1 Pa·s(N·s/m^2) = 10 P(ポアズ)(g/(cm·s))

動粘度　1 m^2/s = 10^4 St(ストークス)(cm^2/s)

圧力

MPa(=10 bar)	kgf/cm^2	atm	mmHg(Torr)	lbf/in^2(psi)
1	10.1972	9.86923	7.50062 × 10^3	145.038
0.0980665	1	0.967841	735.559	14.2233
0.101325	1.03323	1	760	14.6959
1.33322 × 10^{-4}	1.35951 × 10^{-3}	1.31579 × 10^{-3}	1	1.93368 × 10^{-2}
6.89476 × 10^{-3}	7.03070 × 10^{-2}	6.80460 × 10^{-2}	51.7149	1

エネルギー・仕事・熱量

J(=10^7 erg)	kgf·m	kW·h	cal(計量法)	Btu	ft·lbf	eV
1	0.101972	2.77778 × 10^{-7}	0.238889	9.47813 × 10^{-4}	0.737562	6.24150 × 10^{18}
9.80665	1	2.72407 × 10^{-6}	2.34270	9.29487 × 10^{-3}	7.23301	6.12082 × 10^{19}
3.6 × 10^6	3.67098 × 10^5	1	8.59999 × 10^5	3412.13	2.65522 × 10^6	2.24694 × 10^{25}
4.18605	0.426858	1.16279 × 10^{-6}	1	3.96759 × 10^{-3}	3.08747	2.61272 × 10^{19}
1055.06	107.586	2.93072 × 10^{-4}	252.042	1	778.172	6.58515 × 10^{21}
1.35582	0.138255	3.76616 × 10^{-7}	0.323890	1.28506 × 10^{-3}	1	8.46233 × 10^{18}
1.60218 × 10^{-19}	1.63377 × 10^{-20}	4.45050 × 10^{-26}	3.82743 × 10^{-20}	1.51857 × 10^{-22}	1.18171 × 10^{-19}	1

1 cal = 4.18605 J (計量法)
= 4.184 J (熱化学)
= 4.1855 J (15 °C)
= 4.1868 J (国際蒸気表)

仕事率　1 PS (仏馬力)
= 75 kgf·m/s
= 735.499 W

放射能

Bq	Ci
1	2.70270 × 10^{-11}
3.7 × 10^{10}	1

吸収線量

Gy	rad
1	100
0.01	1

照射線量

C/kg	R
1	3876
2.58 × 10^{-4}	1

線量当量

Sv	rem
1	100
0.01	1

（86 年 12 月 26 日現在）

High Energy Particle Transport Code NMTC/JAM