Report BNL-98726-2012-IR
Tech. Note C-AD/AP/470
Oct. 2012

ZGOUBI USERS' GUIDE

ZGOUBI ON WEB :

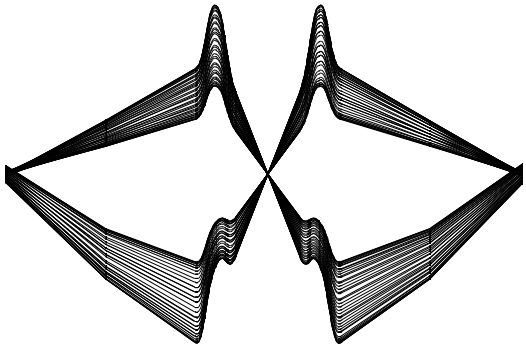
http://sourceforge.net/projects/zgoubi/

François Méot

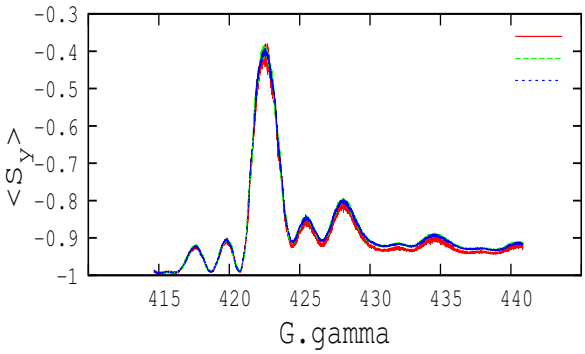
Brookhaven National Laboratory
Collider-Accelerator Department
Upton, NY, 11973

October 5, 2017

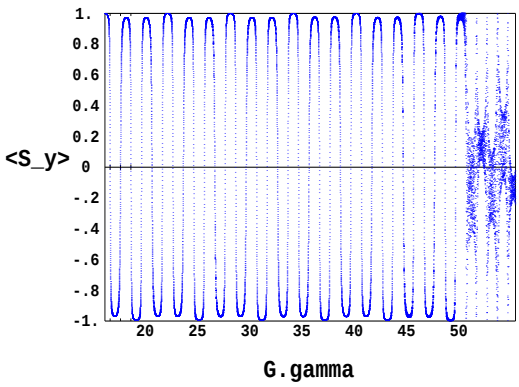
ATLAS & CMS IRs, LHC



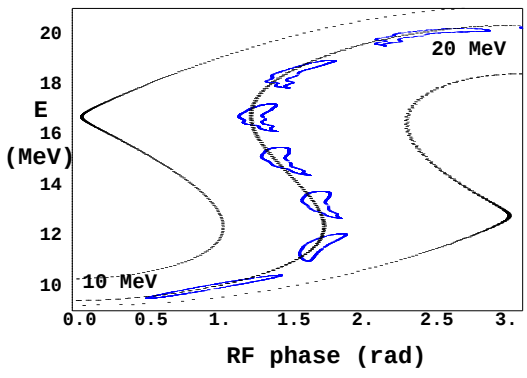
Snake Resonance Crossing in RHIC



Spin flip during ramp, AGS with snakes



Serpentine acceleration in EMMA



Cover figures :
upper left : collision optics at ATLAS and CMS [68],
upper right : evolution of the polarization upon crossing of the 393+Q_y spin resonance in RHIC [14],
lower left : spin-flipping over an acceleration cycle in the AGS, in presence of the helical snakes [13],
lower right : serpentine acceleration in the prototype linear FFAG EMMA [50, 52].

Table of contents

PART A

Description of software contents

Glossary of Keywords, Part A

Available keywords and where they are to be found in Part A.

Optical elements versus keywords

What can be simulated
What keyword(s) can be used for that

This glossary gives a list of keywords suitable for the simulation of common optical elements. These are classified in three categories: magnetic, electric and combined electro-magnetic elements.

Field map procedures are also listed; they provide a means for ray-tracing through measured or simulated electric and/or magnetic fields.

MAGNETIC ELEMENTS

AGS main magnet	AGSMM
Cyclotron magnet or sector	DIPOLE[S], DIPOLE-M, FFAG, FFAG-SPI
Decapole	DECAPOLE, MULTIPOL
Dipole[s], spectrometer dipole	AIMANT, BEND, DIPOLE[S][-M], MULTIPOL, QUADISEX
Dodecapole	DODECAPO, MULTIPOL
FFAG magnet	DIPOLES, FFAG, FFAG-SPI, MULTIPOL
Helical dipole	HELIX
Multipole	MULTIPOL, QUADISEX, SEXQUAD
Octupole	OCTUPOLE, MULTIPOL, QUADISEX, SEXQUAD
Quadrupole	QUADRUPO, MULTIPOL, SEXQUAD, AGSQUAD
Sextupole	SEXTUPOL, MULTIPOL, QUADISEX, SEXQUAD
Skew multipoles	MULTIPOL
Solenoid	SOLENOID
Spectrometer dipole	DIPOLE, DIPOLE-M, DIPOLES
Undulator	UNDULATOR

Using field maps

1-D, cylindrical symmetry	BREVOL
2-D, mid-plane symmetry	CARTEMES, POISSON, TOSCA
2-D, no symmetry	MAP2D
2-D, polar mesh, mid-plane symmetry	POLARMES
3-D, no symmetry	TOSCA
EMMA FFAG quadrupole doublet	EMMA
linear composition of field maps	TOSCA

ELECTRIC ELEMENTS

2-tube (bipotential) lens	EL2TUB
3-tube (unipotential) lens	UNIPOT
Decapole	ELMULT
Dipole	ELMULT
Dodecapole	ELMULT
Multipole	ELMULT
N-electrode mirror/lens, straight slits	ELMIR
N-electrode mirror/lens, circular slits	ELMIRC
Octupole	ELMULT
Quadrupole	ELMULT

R.F. (kick) cavity	CAVITE
Sextupole	ELMULT
Skew multipoles	ELMULT

Using field maps

1-D, cylindrical symmetry	ELREVOL
2-D, no symmetry	MAP2D-E

ELECTRO-MAGNETIC ELEMENTS

Decapole	EBMULT
Dipole	EBMULT
Dodecapole	EBMULT
Multipole	EBMULT
Octupole	EBMULT
Quadrupole	EBMULT
Sextupole	EBMULT
Skew multipoles	EBMULT
Wien filter	SEPARA, WIENFILT

PREFACE TO THE BNL EDITION, 2015+

New functionalities have been introduced :

- *GOTO* keyword, a conditional goto, useful for RLA simulations,
- *INCLUDE* keyword, working like the fortran “include”, more or less equivalent to “call file” as found in other optics codes, useful for long or repetitive sequences.

A consequence of the latter is the possibility of requesting a “zgoubi.SEQUENCE.Out” output file which contains the whole developed optical sequence in a single file.

PREFACE TO THE BNL EDITION, 2012

The previous release of the Zgoubi Users' Guide as a Lab. report dates from 1997, making the present one the latest in a series of five [1]-[4].

zgoubi has undergone substantial developments since the 4th edition of the Users' Guide, in the frame of a number of projects and of beam dynamics studies, as the Neutrino Factory, lepton and hadron colliders, spin dynamics investigations at SuperB, RHIC, etc. As well the list of optical elements has grown, and so did the "Glossary of Keywords" list, pp. 9 and 187, including new simulation and computing procedures, which range from constraint-matching to overlapping magnetic fields capabilities to spin manipulations and other synchrotron radiation damping recipes.

The ouput files zgoubi.fai (local particle data) and zgoubi.plt (step-by-step-particle data across an optical element) have been reformated in columns, making them readily readable by graphic programs as "gnuplot". Ancillary output files, less voluminous than the latter two and useful for producing plots or for functioning checks, with names of the form "zgoubi.*.out" are available from various keywords, usually by specifying "PRINT" in the keyword's arguments, for instance in the case of "*CAVITE*" it will produce "zgoubi.CAVITE.Out" containing time dependence of RF parameters as experienced by the particles at traversal of the cavity, whereas "zgoubi.OPTICS.out" can be obtained from "*OPTICS*" keyword.

This evolution of **zgoubi** allows embedding "*FIT[2]*" procedure within "*REBELOTE*" (**zgoubi**'s DO-loop), whereas the latter has been confered the capability of changing values of any numerical data across zgoubi.dat. This allows parameter scans, while possibly using "*SYSTEM*" to store the outcomes in dedicated external files.

A series of auxiliary computing tools and graphic tools (based on "gnuplot") have been developed, aimed at making the designer's life easier, as, search for closed orbits in periodic machines, computation of optical functions and parameters, tune scans, dynamic aperture scans, spin dynamics data treatment, graphic scripts, etc., and including dedicated ones regarding, *e.g.*, FFAG and cyclotron design, AGS and RHIC studies, synchrotron radiation energy losses and their effects. In addition "python" interfaces are being developed by several users, possibly made available on web by their authors.

The code has been installed on SourceForge with the collaboration of J. S. Berg, in Sept. 2007, there it is fully and freely available [5]. The SourceForge package evolves and is maintained at the pace of on-going projects and design studies. It includes the source files, the post-processor program **zpop**, as well as many examples with their input data files ("zgoubi.dat") and output result files ("zgoubi.res") and the auxiliary toolbox.

The Users' Guide is intended to describe the contents of the most recent version of **zgoubi**. The code and its Guide are both far from being "finished products".

J. S. Berg and N. Tsoupas, BNL, have reviewed the final proof of the present document.

INTRODUCTION TO THE 4th EDITION, 1997

The initial version of **zgoubi**, dedicated to ray-tracing in magnetic fields, was developed by D. Garreta and J.C. Faivre at CEN-Saclay in the early 1970s. It was perfected for the purpose of studying the four spectrometers SPES I, II, III, IV at the Laboratoire National SATURNE (CEA-Saclay, France), and SPEG at GANIL (Caen, France). It is being used since long in several national and foreign laboratories.

The first manual was in French [1]. Accounting for many developments and improvements, and in order to facilitate access to the program an English version of the manual was written at TRIUMF with the assistance of J. Doornbos. P. Stewart prepared the manuscript for publication [2]

An updating of the latter was necessary for accompanying the third version of the code which included developments regarding spin tracking and ray-tracing in combined electric and magnetic fields ; this was done with the help of D. Bunel (SATURNE Laboratory, Saclay) for the preparation of the document and lead to the third release [3].

In the mid-1990s, the computation of synchrotron radiation electromagnetic impulse and spectra was introduced for the purpose of studying interference effects at the LEP synchrotron radiation based diagnostic mini-wiggler. In the mean time, several new optical elements were added, such as electro-magnetic and other electrostatic lenses. Used since several years for special studies in periodic machines (*e.g.*, SATURNE at Saclay, COSY at Julich, LEP and LHC at CERN, Neutrino Factory rings), **zgoubi** has also undergone extensive developments regarding storage ring related features.

These developments of **zgoubi** have strongly benefited of the environment of the Groupe Théorie, Laboratoire National SATURNE, CEA/DSM-Saclay, in the years 1985-1995.

The graphic interface to **zgoubi** (addressed in Part D) has also undergone concomitant extensive developments, which make it a performing tool for the post-processing of **zgoubi** outputs.

The Users' Guide is intended to describe the contents of the most recent version of **zgoubi**, which is far from being a "finished product".

INTRODUCTION

The computer code **zgoubi** calculates trajectories of charged particles in magnetic and electric fields. At the origin specially adapted to the definition and adjustment of beam lines and magnetic spectrometers, it has so evolved that it allows the study of systems including complex sequences of optical elements such as dipoles, quadrupoles, arbitrary multipoles, FFAG magnets and other magnetic or electric devices, and is able as well to handle periodic structures. Compared to other codes, it presents several peculiarities, as follows - a non-exhaustive list :

- a numerical method for integrating the Lorentz equation, based on Taylor series, which optimizes computing time and provides high accuracy and strong symplecticity,
- spin tracking, using the same numerical method as for the Lorentz equation,
- account of stochastic photon emission, and its effects on particle dynamics,
- calculation of the synchrotron radiation electric field and spectra in arbitrary magnetic fields, from the ray-tracing outcomes,
- the possibility of using a mesh, which allows ray-tracing from simulated or measured (1-D, 2-D or 3-D) electric and magnetic field maps,
- a number Monte Carlo procedures : unlimited number of trajectories, in-flight decay, stochastic radiation, etc.,
- built-in fitting procedures allowing arbitrary variables and a large variety of constraints, easily expandable,
- multi-turn tracking in circular accelerators including features proper to machine parameter calculation and survey,
- simulation of time-varying power supplies,
- simulation of arbitrary radio-frequency programs.

The initial version of **zgoubi** was dedicated to ray-tracing in magnetic elements, beam lines, spectrometers. It was perfected for the purpose of studying, and operating, the four spectrometers SPES I, II, III, IV at the Laboratoire National SATURNE (CEA-Saclay, France), and, later, SPEG at GANIL (Caen, France).

Developments regarding spin tracking and ray-tracing in combined electric and magnetic fields were implemented, in the late 1980s and early 1990s respectively.

In the mid-1990s, the computation of synchrotron radiation electromagnetic impulse and spectra was introduced, for the purpose of synchrotron radiation diagnostic R&D at LEP, and further applied to the design of the SR diagnostics installations at LHC in the early 2000s. In the mean time, several new optical elements were added, such as electro-magnetic and other electrostatic lenses. Used since several years for special studies in periodic machines (*e.g.*, SATURNE at Saclay, COSY at Julich, LEP and LHC at CERN), **zgoubi** has also undergone extensive developments regarding storage ring related features.

Many developments have been accomplished since the early 2000s in the frame of a number of project design and beam dynamics studies, as the neutrino factory, lepton and hadron colliders, spin studies at AGS and RHIC, etc. As a consequence the list of optical elements and the compendium of numerical methods, so-called “Glossary of Keywords” list, pp. 9 and 187, has stretched with new simulation and computing procedures, ranging from fitting (*FIT2* procedure ; additional constraints) to overlapping magnetic field capabilities (*DIPOLES*, *FFAG*), spin manipulation (*SPINR*), optical elements (*BEAMBEAM*), radiation damping tools (*SRLOSS*) and many others (*FAISTORE, MAP2D-E, OPTICS, PICKUPS, TWISS*, etc.).

The graphic interface to **zgoubi** (**zpop**, Part D) has undergone extensive developments, making it a convenient companion tool to the use of **zgoubi**.

1 NUMERICAL CALCULATION OF MOTION AND FIELDS

1.1 zgoubi Frame

The reference frame of **zgoubi** is presented in Fig. 1. Its origin is in the median plane on a reference curve which generally (but not always) coincides with the optical axis of optical elements.

1.2 Integration of the Lorentz Equation

The Lorentz equation, which governs the motion of a particle of charge q, relativistic mass m and velocity $\vec{v}$ in electric and magnetic fields $\vec{e}$ and $\vec{b}$, is written

$$\frac{d(m\vec{v})}{dt} = q\,(\vec{e} + \vec{v} \times \vec{b}) \tag{1.2.1}$$

Noting $()' = \dfrac{d()}{ds}$ and taking

$$\vec{u} = \frac{\vec{v}}{v}, \quad ds = v\,dt\ , \quad \vec{u}' = \frac{d\vec{u}}{ds}\ , \quad m\vec{v} = mv\vec{u} = q\,B\rho\,\vec{u} \tag{1.2.2}$$

where $B\rho$ is the rigidity of the particle, this equation can be rewritten

$$(B\rho)'\,\vec{u} + B\rho\,\vec{u}' = \frac{\vec{e}}{v} + \vec{u} \times \vec{b} \tag{1.2.3}$$

From position $\vec{R}(M_0)$ and unit velocity $\vec{u}(M_0)$ at point M_0, position $\vec{R}(M_1)$ and unit velocity $\vec{u}(M_1)$ at point M_1 following a displacement Δs, are obtained from truncated Taylor expansions (Fig. 2)

$$\begin{aligned} \vec{R}(M_1) &\approx \vec{R}(M_0) + \vec{u}(M_0)\,\Delta s + \vec{u}'(M_0)\,\frac{\Delta s^2}{2!} + ... + \vec{u}''''(M_0)\,\frac{\Delta s^6}{6!} \\ \vec{u}(M_1) &\approx \vec{u}(M_0) + \vec{u}'(M_0)\,\Delta s + \vec{u}''(M_0)\,\frac{\Delta s^2}{2!} + ... + \vec{u}''''(M_0)\,\frac{\Delta s^5}{5!} \end{aligned} \tag{1.2.4}$$

The rigidity at M_1 is obtained in the same way from

$$(B\rho)(M_1) \approx (B\rho)(M_0) + (B\rho)'(M_0)\Delta s + ... + (B\rho)''''(M_0)\frac{\Delta s^5}{5!} \tag{1.2.5}$$

The equation of time of flight is written in a similar manner

$$T(M_1) \approx T(M_0) + T'(M_0)\,\Delta s + T''(M_0)\,\frac{\Delta s^2}{2} + \ ...\ + T''''(M_0)\,\frac{\Delta s^5}{5!} \tag{1.2.6}$$

The derivatives $\vec{u}^{(n)} = \dfrac{d^n\vec{u}}{ds^n}$ and $(B\rho)^{(n)} = \dfrac{d^n(B\rho)}{ds^n}$ involved in these expressions are calculated as described in the next sections. For the sake of computing speed, three distinct software procedures are involved, depending on whether $\vec{e}$ or $\vec{b}$ is zero, or $\vec{e}$ and $\vec{b}$ are both non-zero.

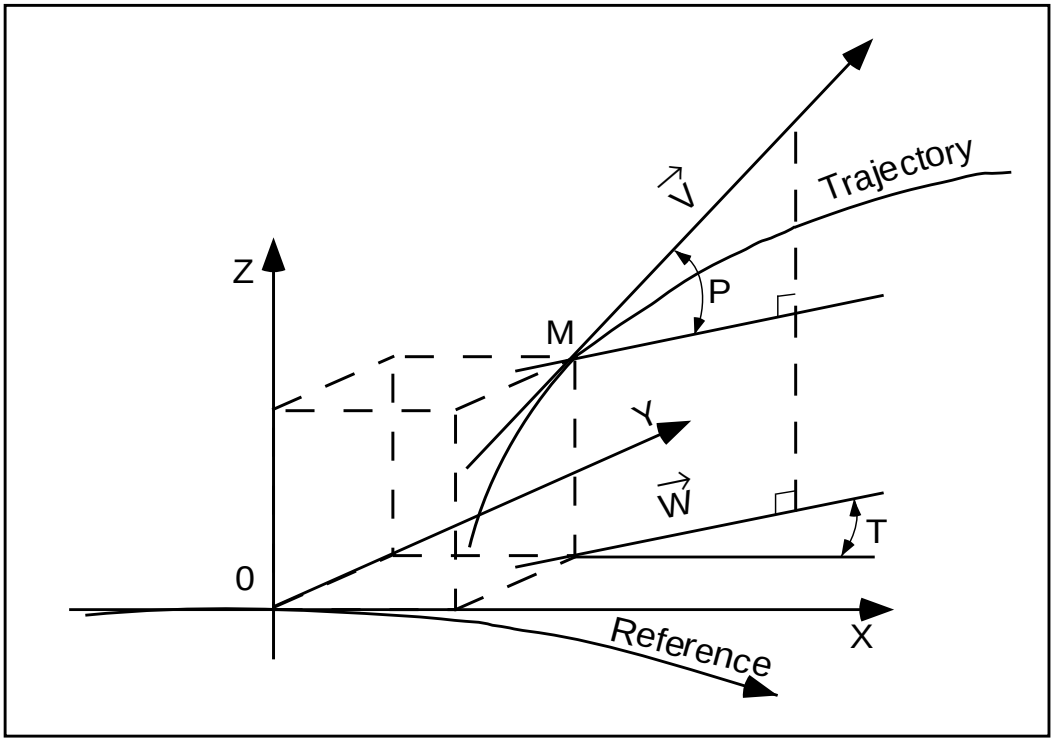


Figure 1: Reference frame and coordinates (Y, T, Z, P) in **zgoubi**.
OX : in the direction of motion,
OY : normal to OX,
OZ : orthogonal to the (X, Y) plane,
$\vec{W}$: projection of the velocity, $\vec{v}$, in the (X, Y) plane,
T = angle between $\vec{W}$ and the X-axis,
P = angle between $\vec{W}$ and $\vec{v}$.

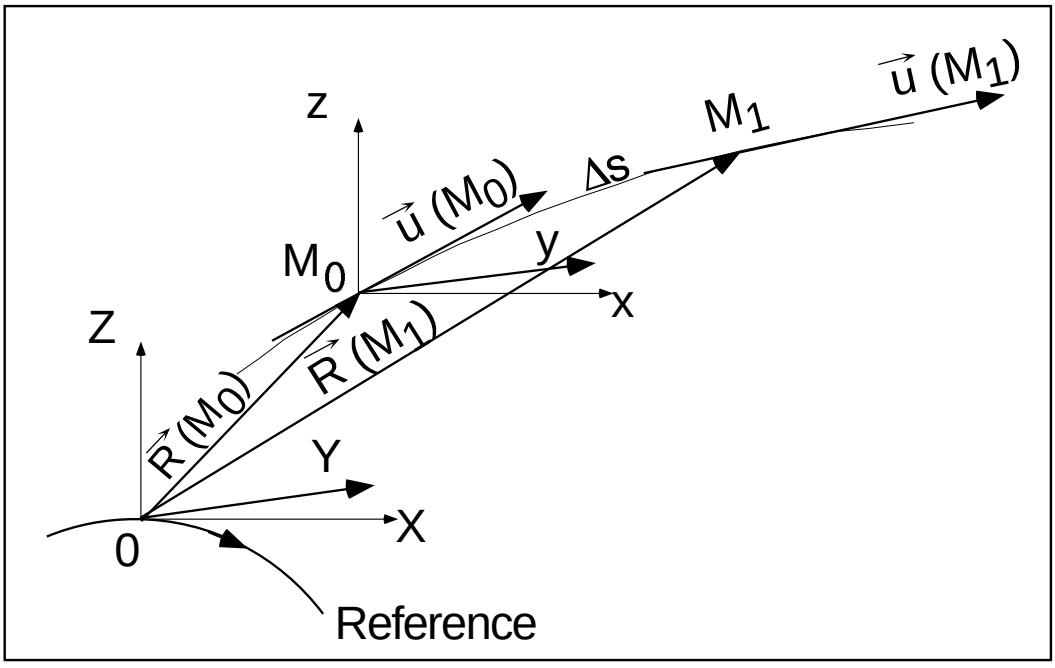


Figure 2: Position and velocity of a particle in the reference frame.

1.2.1 Integration in Magnetic Fields

Considering $\vec{e} = 0$ and given $\vec{B} = \dfrac{\vec{b}}{B\rho}$, eq. (1.2.3) reduces to

$$\vec{u}' = \vec{u} \times \vec{B} \tag{1.2.7}$$

The successive derivatives $\vec{u}^{(n)} = \dfrac{d^n\vec{u}}{ds^n}$ of $\vec{u}$ needed in the Taylor expansions (eqs. 1.2.4) are calculated by differentiating $\vec{u}' = \vec{u} \times \vec{B}$

$$\begin{aligned}
\vec{u}'' &= \vec{u}' \times \vec{B} + \vec{u} \times \vec{B}' \\
\vec{u}''' &= \vec{u}'' \times \vec{B} + 2\vec{u}' \times \vec{B}' + \vec{u} \times \vec{B}'' \\
\vec{u}'''' &= \vec{u}''' \times \vec{B} + 3\vec{u}'' \times \vec{B}' + 3\vec{u}' \times \vec{B}'' + \vec{u} \times \vec{B}''' \\
\vec{u}''''' &= \vec{u}'''' \times \vec{B} + 4\vec{u}''' \times \vec{B}' + 6\vec{u}'' \times \vec{B}'' + 4\vec{u}' \times \vec{B}''' + \vec{u} \times \vec{B}''''
\end{aligned} \tag{1.2.8}$$

where $\vec{B}^{(n)} = \dfrac{d^n\vec{B}}{ds^n}$.
From

$$d\vec{B} = \frac{\partial\vec{B}}{\partial X}\,dX + \frac{\partial\vec{B}}{\partial Y}\,dY + \frac{\partial\vec{B}}{\partial Z}\,dZ = \sum_{i=1,3} \frac{\partial\vec{B}}{\partial X_i}\,dX_i \tag{1.2.9}$$

and by successive differentiation, we get

$$\begin{aligned}
\vec{B}' &= \sum_i \frac{\partial\vec{B}}{\partial X_i}\,u_i \\
\vec{B}'' &= \sum_{ij} \frac{\partial^2\vec{B}}{\partial X_i \partial X_j}\,u_i u_j + \sum_i \frac{\partial\vec{B}}{\partial X_i}\,u_i' \\
\vec{B}''' &= \sum_{ijk} \frac{\partial^3\vec{B}}{\partial X_i \partial X_j \partial X_k}\,u_i u_j u_k + 3\sum_{ij} \frac{\partial^2\vec{B}}{\partial X_i \partial X_j}\,u_i' u_j + \sum_i \frac{\partial\vec{B}}{\partial X_i}\,u_i'' \\
\vec{B}'''' &= \sum_{ijkl} \frac{\partial^4\vec{B}}{\partial X_i \partial X_j \partial X_k \partial X_l}\,u_i u_j u_k u_l + 6\sum_{ijk} \frac{\partial^3\vec{B}}{\partial X_i \partial X_j \partial X_k}\,u_i' u_j u_k \\
&\quad + 4\sum_{ij} \frac{\partial^2\vec{B}}{\partial X_i \partial X_j}\,u_i'' u_j + 3\sum_{ij} \frac{\partial^2\vec{B}}{\partial X_i \partial X_j}\,u_i' u_j' + \sum_i \frac{\partial\vec{B}}{\partial X_i}\,u_i'''
\end{aligned} \tag{1.2.10}$$

From the knowledge of $\vec{u}(M_0)$ and $\vec{B}(M_0)$ at point M_0 of the trajectory, we calculate alternately the derivatives of $\vec{u}(M_0)$ and $\vec{B}(M_0)$, by means of eqs. (1.2.8) and (1.2.10), and inject these into eq. (1.2.4) so yielding $\vec{R}(M_1)$ and $\vec{u}(M_1)$.

1.2.2 Integration in Electric Fields [6]

Admitting that $\vec{b} = 0$, eq. (1.2.3) reduces to

$$(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} \tag{1.2.11}$$

which, by successive differentiations, gives the recursive relations

$$
\begin{aligned}
&(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} \\
&(B\rho)''\vec{u} + 2(B\rho)'\vec{u}' + B\rho\vec{u}'' = \left(\frac{1}{v}\right)'\vec{e} + \frac{\vec{e}'}{v} \\
&(B\rho)'''\vec{u} + 3(B\rho)''\vec{u}' + 3(B\rho)'\vec{u}'' + B\rho\vec{u}''' = \left(\frac{1}{v}\right)''\vec{e} + 2\left(\frac{1}{v}\right)'\vec{e}' + \left(\frac{1}{v}\right)\vec{e}'' \\
&(B\rho)''''\vec{u} + 4(B\rho)'''\vec{u}' + 6(B\rho)''\vec{u}'' + 4(B\rho)'\vec{u}''' + B\rho\vec{u}'''' = \\
&\qquad \left(\frac{1}{v}\right)'''\vec{e} + 3\left(\frac{1}{v}\right)''\vec{e}' + 3\left(\frac{1}{v}\right)'\vec{e}'' + \frac{1}{v}\vec{e}''' \\
&(B\rho)'''''\vec{u} + 5(B\rho)''''\vec{u}' + \ etc.
\end{aligned}
\tag{1.2.12}
$$

that provide the derivatives $\dfrac{d^n\vec{u}}{ds^n}$ needed in the Taylor expansions (eq. 1.2.4)

$$
\begin{aligned}
\vec{u}' &= \left(\frac{1}{v}\right)\vec{E} - \frac{(B\rho)'}{B\rho}\vec{u} \\
\vec{u}'' &= \left(\frac{1}{v}\right)'\vec{E} + \left(\frac{1}{v}\right)\vec{E}'|_{B\rho} - 2\frac{(B\rho)'}{B\rho}\vec{u}' - \frac{(B\rho)''}{B\rho}\vec{u} \\
\vec{u}''' &= \left(\frac{1}{v}\right)''\vec{E} + 2\left(\frac{1}{v}\right)'\vec{E}'|_{B\rho} + \frac{1}{v}\vec{E}''|_{B\rho} - 3\frac{(B\rho)'}{B\rho}\vec{u}'' - 3\frac{(B\rho)''}{B\rho}\vec{u}' - \frac{(B\rho)'''}{B\rho}\vec{u} \\
\vec{u}'''' &= \left(\frac{1}{v}\right)'''\vec{E} + 3\left(\frac{1}{v}\right)''\vec{E}'|_{B\rho} + 3\left(\frac{1}{v}\right)'\vec{E}''|_{B\rho} + \left(\frac{1}{v}\right)\vec{E}'''|_{B\rho} \\
&\quad - 4\frac{(B\rho)'}{B\rho}\vec{u}''' - 6\frac{(B\rho)''}{B\rho}\vec{u}'' - 4\frac{(B\rho)'''}{B\rho}\vec{u}' - \frac{(B\rho)''''}{B\rho}\vec{u} \\
\vec{u}''''' &= \left(\frac{1}{v}\right)''''\vec{E} + 4\left(\frac{1}{v}\right)'''\vec{E}'|_{B\rho} + 6\left(\frac{1}{v}\right)''\vec{E}''|_{B\rho} + 4\left(\frac{1}{v}\right)'\vec{E}'''|_{B\rho} + \left(\frac{1}{v}\right)\vec{E}''''|_{B\rho} \\
&\quad - 5\frac{(B\rho)'}{B\rho}\vec{u}'''' - 10\frac{(B\rho)''}{B\rho}\vec{u}''' - 10\frac{(B\rho)'''}{B\rho}\vec{u}'' - 5\frac{(B\rho)''''}{B\rho}\vec{u}' - \frac{(B\rho)'''''}{B\rho}\vec{u}
\end{aligned}
\tag{1.2.13}
$$

where $\vec{E} = \dfrac{\vec{e}}{B\rho}$, and $(\ )^{(n)}|_{B\rho}$ denotes differentiation at constant $B\rho$: $\vec{E}^{(n)}|_{B\rho} = \dfrac{1}{B\rho}\dfrac{d^n\vec{e}}{ds^n}$. These derivatives of the electric field are obtained from the total derivative

$$
d\vec{E} = \frac{\partial\vec{E}}{\partial X}\,dX + \frac{\partial\vec{E}}{\partial Y}\,dY + \frac{\partial\vec{E}}{\partial Z}\,dZ \tag{1.2.14}
$$

by successive differentiations

$$\begin{aligned}
\vec{E}' &= \sum_i \frac{\partial \vec{E}}{\partial X_i}\, u_i \\
\vec{E}'' &= \sum_{ij} \frac{\partial^2 \vec{E}}{\partial X_i \partial X_j}\, u_i u_j + \sum_i \frac{\partial \vec{E}}{\partial X_i}\, u_i' \\
\vec{E}''' &= \sum_{ijk} \frac{\partial^3 \vec{E}}{\partial X_i \partial X_j \partial X_k}\, u_i u_j u_k + 3 \sum_{ij} \frac{\partial^2 \vec{E}}{\partial X_i \partial X_j}\, u_i' u_j + \sum_i \frac{\partial \vec{E}}{\partial X_i}\, u_i''
\end{aligned} \tag{1.2.15}$$

etc. as in eq. 1.2.10. The eqs. (1.2.13), as well as the calculation of the rigidity, eq. (1.2.5), involve derivatives $(B\rho)^{(n)} = \dfrac{d^n(B\rho)}{ds^n}$, which are obtained in the following way. Considering that

$$\frac{dp^2}{dt} = \frac{d\vec{p}^2}{dt} \quad \textit{i.e.,} \quad \frac{dp}{dt}\, p = \frac{d\vec{p}}{dt}\vec{p} \tag{1.2.16}$$

with $\dfrac{d\vec{p}}{dt} = q\,(\vec{e} + \vec{v} \times \vec{b})$ (eq. 1.2.1), we obtain

$$\frac{dp}{dt}\, p = q\,(\vec{e} + v \times \vec{b}) \cdot \vec{p} = q\vec{e} \cdot \vec{p} \tag{1.2.17}$$

since $(\vec{v} \times \vec{b}) \cdot \vec{p} = 0$. Normalizing as previously with $\vec{p} = p\vec{u} = qB\rho\vec{u}$ and $ds = vdt$, and by successive differentiations, eq. (1.2.17) leads to the $(B\rho)^{(n)}$

$$\begin{aligned}
(B\rho)' &= \frac{1}{v}\,(\vec{e} \cdot \vec{u}) \\
(B\rho)'' &= \left(\frac{1}{v}\right)'(\vec{e} \cdot \vec{u}) + \frac{1}{v}\,(\vec{e} \cdot \vec{u})' \\
(B\rho)''' &= \left(\frac{1}{v}\right)''(\vec{e} \cdot \vec{u}) + 2\left(\frac{1}{v}\right)'(\vec{e} \cdot \vec{u})' + \frac{1}{v}\,(\vec{e} \cdot \vec{u})'' \\
(B\rho)'''' &= \left(\frac{1}{v}\right)'''(\vec{e} \cdot \vec{u}) + 3\left(\frac{1}{v}\right)''(\vec{e} \cdot \vec{u})' + 3\left(\frac{1}{v}\right)'(\vec{e} \cdot \vec{u})'' + \frac{1}{v}\,(\vec{e} \cdot \vec{u})''' \\
(B\rho)''''' &= \; etc.
\end{aligned} \tag{1.2.18}$$

The derivatives $(\vec{e} \cdot \vec{u})^{(n)} = \dfrac{d^n(\vec{e} \cdot \vec{u})}{ds^n}$ are obtained by recursive differentiation

$$\begin{aligned}
(\vec{e} \cdot \vec{u})' &= \vec{e}' \cdot \vec{u} + \vec{e} \cdot \vec{u}' \\
(\vec{e} \cdot \vec{u})'' &= \vec{e}'' \cdot \vec{u} + 2\,\vec{e}' \cdot \vec{u}' + \vec{e} \cdot \vec{u}'' \\
&\quad \text{etc.}
\end{aligned} \tag{1.2.19}$$

Note that they can be related to the derivatives of the kinetic energy W by $dW = \dfrac{d\vec{p}}{dt} \cdot \vec{v}\, dt = q\vec{e} \cdot \vec{v}\, dt$ which leads to

$$\frac{d^{n+1}W}{ds^{n+1}} = q\frac{d^n(\vec{e}\cdot\vec{u})}{ds^n} \tag{1.2.20}$$

Finally, the derivatives $\left(\frac{1}{v}\right)^{(n)} = \frac{d^n\left(\frac{1}{v}\right)}{ds^n}$ involved in eqs. (1.2.13,1.2.18) are obtained from

$$p = \frac{v}{c}\frac{W + m_0c^2}{c}$$

(m_0 is the rest mass) by successive differentiations, that give the recursive relations

$$\begin{aligned}
\left(\frac{1}{v}\right) &= \frac{1}{c^2}\,\frac{W + m_0c^2}{qB\rho} \\
\left(\frac{1}{v}\right)' &= \frac{1}{c^2}\,\frac{(\vec{e}\cdot\vec{u})}{B\rho} - \frac{1}{v}\frac{(B\rho)'}{B\rho} \\
\left(\frac{1}{v}\right)'' &= \frac{1}{c^2}\,\frac{(\vec{e}\cdot\vec{u})'}{B\rho} - 2\left(\frac{1}{v}\right)'\frac{(B\rho)'}{B\rho} - \frac{1}{v}\,\frac{(B\rho)''}{B\rho} \\
\left(\frac{1}{v}\right)''' &= \frac{1}{c^2}\,\frac{(\vec{e}\cdot\vec{u})''}{B\rho} - 3\left(\frac{1}{v}\right)''\frac{(B\rho)'}{B\rho} - 3\left(\frac{1}{v}\right)'\frac{(B\rho)''}{B\rho} - \frac{1}{v}\,\frac{(B\rho)'''}{B\rho} \\
\left(\frac{1}{v}\right)'''' &= \ etc.
\end{aligned} \tag{1.2.21}$$

1.2.3 Integration in Combined Electric and Magnetic Fields

When both $\vec{e}$ and $\vec{b}$ are non-zero, the complete eq. (1.2.3) must be considered. Recursive differentiations give the following relations

$$\begin{aligned}
&(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} + \vec{u}\times\vec{b} \\
&(B\rho)''\vec{u} + 2(B\rho)'\vec{u}' + B\rho\vec{u}'' = \left(\frac{1}{v}\right)'\vec{e} + \left(\frac{1}{v}\right)\vec{e}' + (\vec{u}\times\vec{b})' \\
&(B\rho)'''\vec{u} + 3(B\rho)''\vec{u}' + 3(B\rho)'\vec{u}'' + B\rho\vec{u}''' = \left(\frac{1}{v}\right)''\vec{e} + 2\left(\frac{1}{v}\right)'\vec{e}' + \left(\frac{1}{v}\right)\vec{e}'' + (\vec{u}\times\vec{b})'' \\
&(B\rho)''''\vec{u} + 4(B\rho)'''\vec{u}' + 6(B\rho)''\vec{u}'' + 4(B\rho)'\vec{u}''' + B\rho\vec{u}'''' = \\
&\qquad \left(\frac{1}{v}\right)'''\vec{e} + 3\left(\frac{1}{v}\right)''\vec{e}' + 3\left(\frac{1}{v}\right)'\vec{e}'' + \frac{1}{v}\vec{e}'''' + (\vec{u}\times\vec{b})'''
\end{aligned} \tag{1.2.22}$$

that provide the derivatives $\frac{d^n\vec{u}}{ds^n}$ needed in the Taylor expansions (1.2.4)

$$
\begin{aligned}
\vec{u}' &= \left(\frac{1}{v}\right)\vec{E} + (\vec{u}\times\vec{B}) - \frac{(B\rho)'}{B\rho}\vec{u} \\
\vec{u}'' &= \left(\frac{1}{v}\right)'\vec{E} + \left(\frac{1}{v}\right)\vec{E}'|_{B\rho} + (\vec{u}\times\vec{B})'|_{B\rho} - 2\frac{(B\rho)'}{B\rho}\vec{u}' - \frac{(B\rho)''}{B\rho}\vec{u} \\
\vec{u}''' &= \left(\frac{1}{v}\right)''\vec{E} + 2\left(\frac{1}{v}\right)'\vec{E}'|_{B\rho} + \frac{1}{v}\vec{E}''|_{B\rho} + (\vec{u}\times\vec{B})''|_{B\rho} - 3\frac{(B\rho)'}{B\rho}\vec{u}'' - 3\frac{(B\rho)''}{B\rho}\vec{u}' - \frac{(B\rho)'''}{B\rho}\vec{u} \\
\vec{u}'''' &= \left(\frac{1}{v}\right)'''\vec{E} + 3\left(\frac{1}{v}\right)''\vec{E}'|_{B\rho} + 3(\frac{1}{v})'\vec{E}''|_{B\rho} + \left(\frac{1}{v}\right)\vec{E}'''|_{B\rho} \\
&\quad + (\vec{u}\times\vec{B})'''|_{B\rho} - 4\frac{(B\rho)'}{B\rho}\vec{u}''' - 6\frac{(B\rho)''}{B\rho}\vec{u}'' - 4\frac{(B\rho)'''}{B\rho}\vec{u}' - \frac{(B\rho)''''}{B\rho}\vec{u}
\end{aligned}
\tag{1.2.23}
$$

where $\vec{E} = \dfrac{\vec{e}}{B\rho}$, $\vec{B} = \dfrac{\vec{b}}{B\rho}$, and $^{(n)}|_{B\rho}$ denotes differentiation at constant $B\rho$

$$
\vec{E}^{(n)}|_{B\rho} = \frac{1}{B\rho}\frac{d^n\vec{e}}{ds^n} \quad \text{and} \quad (\vec{u}\times\vec{B})^{(n)}|_{B\rho} = \frac{1}{B\rho}(\vec{u}\times\vec{b})^{(n)}. \tag{1.2.24}
$$

These derivatives $\vec{E}^{(n)}$ and $\vec{B}^{(n)}$ of the electric and magnetic fields are calculated from the vector fields $\vec{E}(X,Y,Z)$, $\vec{B}(X,Y,Z)$ and their derivatives $\dfrac{\partial^{i+j+k}\vec{E}}{\partial X^i\partial Y^j\partial Z^k}$ and $\dfrac{\partial^{i+j+k}\vec{B}}{\partial X^i\partial Y^j\partial Z^k}$, following eqs. (1.2.14, 1.2.15) and eqs. (1.2.9, 1.2.10), respectively.

1.2.4 Calculation of the Time of Flight

The time of flight eq. (1.2.6) involves the derivatives $dT/ds = 1/v$, $d^2T/ds^2 = d(1/v)/ds$, etc. that are obtained from eq. (1.2.21). In the absence of electric field, however, eq. (1.2.6) reduces to the simple form

$$
T(M_1) = T(M_0) + \Delta s/v \tag{1.2.25}
$$

1.3 Calculation of $\vec{E}$ and $\vec{B}$ Fields and their Derivatives

In this section, unless otherwise stated, $\vec{B} = (B_X(X,Y,Z), B_Y(X,Y,Z), B_Z(X,Y,Z))$ stands indifferently for electric field $\vec{E}$ or magnetic field $\vec{B}$.

$\vec{B}(X,Y,Z)$ and derivatives are calculated in various ways, depending whether field maps or analytic representations of optical elements are used. The basic means are the following.

1.3.1 1-D (Axial) Analytical Field Models and Extrapolation

This procedure assumes cylindrical symmetry with respect to the X-axis. The longitudinal field component B_X or $E_X(X, r = 0)$ $(r = (Y^2 + Z^2)^{1/2})$, along that axis is derived by differentiation of an appropriate model of the potential $V(X)$ (e.g., magnetic in *SOLENOID*, electrostatic in *EL2TUB, UNIPOT*). The longitudinal and radial field components $B_X(X, r)$, $B_r(X, r)$ and their derivatives off-axis $\dfrac{\partial^{i+j} B_X}{\partial X^i \partial r^j}$ and $\dfrac{\partial^{i+j} B_r}{\partial X^i \partial r^j}$ are obtained by Taylor expansions to the second to fifth order in r (depending on the optical element) assuming cylindrical symmetry (eq. (1.3.1)), and then transformed to the (X, Y, Z) Cartesian frame of **zgoubi** in order to provide the derivatives $\dfrac{\partial^{i+j+k} \vec{B}}{\partial X^i \partial Y^j \partial Z^k}$ needed in eq. (1.2.15).

1.3.2 Extrapolation from 1-D axial field map [7]

A cylindrically symmetric field (*e.g.*, using *BREVOL*, *ELREVOL*) can be described by an axial 1-D field map of its longitudinal component $B_X(X, r = 0)$ $(r = (Y^2 + Z^2)^{1/2})$, while the radial component on axis $B_r(X, r = 0)$ is assumed to be zero. $B_X(X, r = 0)$ is obtained at any point along the X-axis by a polynomial interpolation from the map mesh (see section 1.4.1). Then the field components $B_X(X, r)$, $B_r(X, r)$ at the position of the particle, (X, r) are obtained from Taylor expansions truncated at the fifth order in r (hence, up to the fifth order derivative $\dfrac{\partial^5 B_X}{\partial X^5}(X, 0)$), assuming cylindrical symmetry

$$\begin{aligned} B_X(X,r) &= B_X(X,0) - \frac{r^2}{4}\frac{\partial^2 B_X}{\partial X^2}(X,0) + \frac{r^4}{64}\frac{\partial^4 B_X}{\partial X^4}(X,0) \\ B_r(X,r) &= -\frac{r}{2}\frac{\partial B_X}{\partial X}(X,0) + \frac{r^3}{16}\frac{\partial^3 B_X}{\partial X^3}(X,0) - \frac{r^5}{384}\frac{\partial^5 B_X}{\partial X^5}(X,0) \end{aligned} \tag{1.3.1}$$

Then, by differentiation with respect to X and r, up to the second order, these expressions provide the derivatives of $\vec{B}(X, r)$. Finally a conversion from the (X, r) coordinates to the (X, Y, Z) Cartesian coordinates of **zgoubi** is performed, thus providing the expressions $\dfrac{\partial^{i+j+k} \vec{B}}{\partial X^i \partial Y^j \partial Z^k}$ needed in the eq. (1.2.10).

1.3.3 Extrapolation From Median Plane Field Models

In the median plane, $B_Z(X, Y, 0)$ and its derivatives with respect to X or Y may be derived from analytical models (*e.g.*, in Venus magnet - *VENUS*, and sharp edge multipoles *SEXQUAD* and *QUADISEX*) or numerically by polynomial interpolation from 2-D field maps (*e.g.*, *CARTEMES, TOSCA*).
Median plane antisymmetry is assumed, which results in

$$\begin{aligned} B_X(X,Y,0) &= 0 \\ B_Y(X,Y,0) &= 0 \\ B_X(X,Y,Z) &= -B_X(X,Y,-Z) \\ B_Y(X,Y,Z) &= -B_Y(X,Y,-Z) \\ B_Z(X,Y,Z) &= B_Z(X,Y,-Z) \end{aligned} \tag{1.3.2}$$

Accommodated with Maxwell's equations, this results in Taylor expansions below, for the three components of $\vec{B}$ (here, B stands for $B_Z(X, Y, 0)$)

$$
\begin{aligned}
B_X(X,Y,Z) &= Z\frac{\partial B}{\partial X} - \frac{Z^3}{6}\left(\frac{\partial^3 B}{\partial X^3} + \frac{\partial^3 B}{\partial X \partial Y^2}\right) \\
B_Y(X,Y,Z) &= Z\frac{\partial B}{\partial Y} - \frac{Z^3}{6}\left(\frac{\partial^3 B}{\partial X^2 \partial Y} + \frac{\partial^3 B}{\partial Y^3}\right) \\
B_Z(X,Y,Z) &= B - \frac{Z^2}{2}\left(\frac{\partial^2 B}{\partial X^2} + \frac{\partial^2 B}{\partial Y^2}\right) + \frac{Z^4}{24}\left(\frac{\partial^4 B}{\partial X^4} + 2\frac{\partial^4 B}{\partial X^2 \partial Y^2} + \frac{\partial^4 B}{\partial Y^4}\right)
\end{aligned}
\tag{1.3.3}
$$

which are then differentiated one by one with respect to X, Y, or Z, up to second or fourth order (depending on optical element or *IORDRE* option, see section 1.4.2) so as to get the expressions involved in eq. (1.2.10).

1.3.4 Extrapolation from Arbitrary 2-D Field Maps

2-D field maps that give the three components $B_X(X,Y,Z_0)$, $B_Y(X,Y,Z_0)$ and $B_Z(X,Y,Z_0)$ at each node (X,Y) of a Z_0 Z-elevation map may be used. $\vec{B}$ and its derivatives at any point (X,Y,Z) are calculated by polynomial interpolation followed by Taylor expansions in Z, without any hypothesis of symmetries (see section 1.4.3 and keywords *MAP2D, MAP2D-E*).

1.3.5 Interpolation in 3-D Field Maps [8]

In 3-D field maps $\vec{B}$ and its derivatives up to the second order with respect to X, Y or Z are calculated by means of a second order polynomial interpolation, from 3-D $3 \times 3\times$ 3-point grid (see section 1.4.4).

1.3.6 2-D Analytical Field Models and Extrapolation

Several optical elements such as *BEND, WIENFILT* (that uses the *BEND* procedures), *QUADISEX, VENUS*, etc., are defined from the expression of the field and derivatives in the median plane. 3-D extrapolation of these off the median plane is drawn from Taylor expansions and Maxwell's equations.

1.3.7 3-D Analytical Models of Fields

In many optical elements such as *QUADRUPO, SEXTUPOL, MULTIPOL, EBMULT*, etc., the three components of $\vec{B}$ and their derivatives with respect to X, Y or Z are obtained at any step along trajectories from analytical expression drawn from the scalar potential $V(X,Y,Z)$, namely

$$
B_X = \frac{\partial V}{\partial X}, \quad B_Y = \frac{\partial V}{\partial Y}, \quad B_Z = \frac{\partial V}{\partial Z}, \quad \frac{\partial B_X}{\partial X} = \frac{\partial^2 V}{\partial X^2}, \quad \frac{\partial B_X}{\partial Y} = \frac{\partial^2 V}{\partial X \partial Y}, \quad \text{etc.} \tag{1.3.4}
$$

and similarly for $\vec{E}$ with opposite sign for the gradients.

Multipoles

The scalar potential used for the calculation of $\dfrac{\partial^{i+j+k}\vec{B}_n(X,Y,Z)}{\partial X^i \partial Y^i \partial Z^k}$ ($i+j+k = 0$ to 4) in the case of magnetic and electro-magnetic multipoles with $2n$ poles (namely, *QUADRUPO* ($n=2$) to *DODECAPO* ($n=6$), *MULTIPOL* ($n=1$ to 10), *EBMULT* ($n=1$ to 10)) is [9]

$$V_n(X,Y,Z) = (n!)^2 \left(\sum_{q=0}^{\infty} (-1)^q \frac{G^{(2q)}(X)(Y^2+Z^2)^q}{4^q q!(n+q)!} \right) \left(\sum_{m=0}^{n} \frac{\sin\left(m\frac{\pi}{2}\right) Y^{n-m} Z^m}{m!(n-m)!} \right) \tag{1.3.5}$$

where $G(X)$ is a longitudinal form factor, defined at the entrance or exit of the optical element by

$$G(s) = \frac{G_0}{1+\exp(P(s))}, \quad G_0 = \frac{B_0}{R_0^{n-1}} \tag{1.3.6}$$

wherein B_0 is the field at pole tip radius R_0, and

$$P(s) = C_0 + C_1\left(\frac{s}{\lambda}\right) + C_2\left(\frac{s}{\lambda}\right)^2 + C_3\left(\frac{s}{\lambda}\right)^3 + C_4\left(\frac{s}{\lambda}\right)^4 + C_5\left(\frac{s}{\lambda}\right)^5$$

and s is the distance to the EFB.

Skew Multipoles

A multipole component with arbitrary order n can be tilted independently of the others by an arbitrary angle A_n around the X-axis. If so, the calculation of the field and derivatives in the rotated axis (X, Y_R, Z_R) is done in two steps. First, they are calculated at the rotated position (X, Y_R, Z_R), in the (X, Y, Z) frame, using the expression (1.3.5) above. Second, $\vec{B}$ and its derivatives at (X, Y_R, Z_R) in the (X, Y, Z) frame are transformed into the new, (X, Y_R, Z_R) frame, by a rotation with angle A_n.

In particular a skew $2n$-pole component is created by taking $A_n = \pi/2n$.

A Note on Electrostatic Multipoles

A right electric multipole has the same field equations as the like-order skew magnetic multipole. Therefore, calculation of right or skew electric or electro-magnetic multipoles (*ELMULT, EBMULT, ELMULT*) uses the same eq. (1.3.5) together with the rotation process as described in section 1.3.7. The same method is used for arbitrary rotation of any multipole component around the X-axis.

1.4 Calculation of $\vec{E}$ and $\vec{B}$ from Field Maps

In this section, unless otherwise stated, $\vec{B} = (B_X(X,Y,Z), B_Y(X,Y,Z), B_Z(X,Y,Z))$ stands indifferently for electric field $\vec{E}$ or magnetic field $\vec{B}$.

1.4.1 1-D Axial Map, with Cylindrical Symmetry

Let B_i be the value of the longitudinal component $B_X(X, r=0)$ of the field $\vec{B}$, at node i of a uniform mesh that defines a 1-D field map along the symmetry X-axis, while $B_r(X, r=0)$ is assumed to be zero $(r = (Y^2+Z^2)^{1/2})$. The field component $B_X(X, r=0)$ is calculated by a polynomial interpolation of the fifth degree in X, using a 5 points grid centered at the node of the 1-D map which is closest to the actual coordinate X of the particle.
The interpolation polynomial is

$$B(X,0) = A_0 + A_1X + A_2X^2 + A_3X^3 + A_4X^4 + A_5X^5 \tag{1.4.1}$$

and the coefficients A_i are calculated by expressions that minimize the quadratic sum

$$S = \sum_i (B(X,0) - B_i)^2 \tag{1.4.2}$$

Namely, the source code contains the explicit analytical expressions of the coefficients A_i solutions of the normal equations $\partial S/\partial A_i = 0$.

The derivatives $\dfrac{\partial^n B}{\partial X^n}(X,0)$ at the actual position X, as involved in eqs. (1.3.1), are then obtained by differentiation of the polynomial (1.4.1), giving

$$\begin{aligned}
\frac{\partial B}{\partial X}(X,0) &= A_1 + 2A_2X + 3A_3X^2 + 4A_4X^3 + 5A_5X^4 \\
\frac{\partial^2 B}{\partial X^2}(X,0) &= 2A_2 + 6A_3X + 12A_4X^2 + 20A_5X^3 \\
&\cdots \\
\frac{\partial^5 B}{\partial X^5}(X,0) &= 120A_5
\end{aligned} \tag{1.4.3}$$

1.4.2 2-D Median Plane Map, with Median Plane Antisymmetry

Let B_{ij} be the value of $B_Z(X,Y,0)$ at the nodes of a mesh which defines a 2-D field map in the (X,Y) plane while $B_X(X,Y,0)$ and $B_Y(X,Y,0)$ are assumed to be zero. Such a map may have been built or measured in either Cartesian or polar coordinates. Whenever polar coordinates are used, a change to Cartesian coordinates (described below) provides the expression of $\vec{B}$ and its derivatives as involved in eq. (1.2.10).

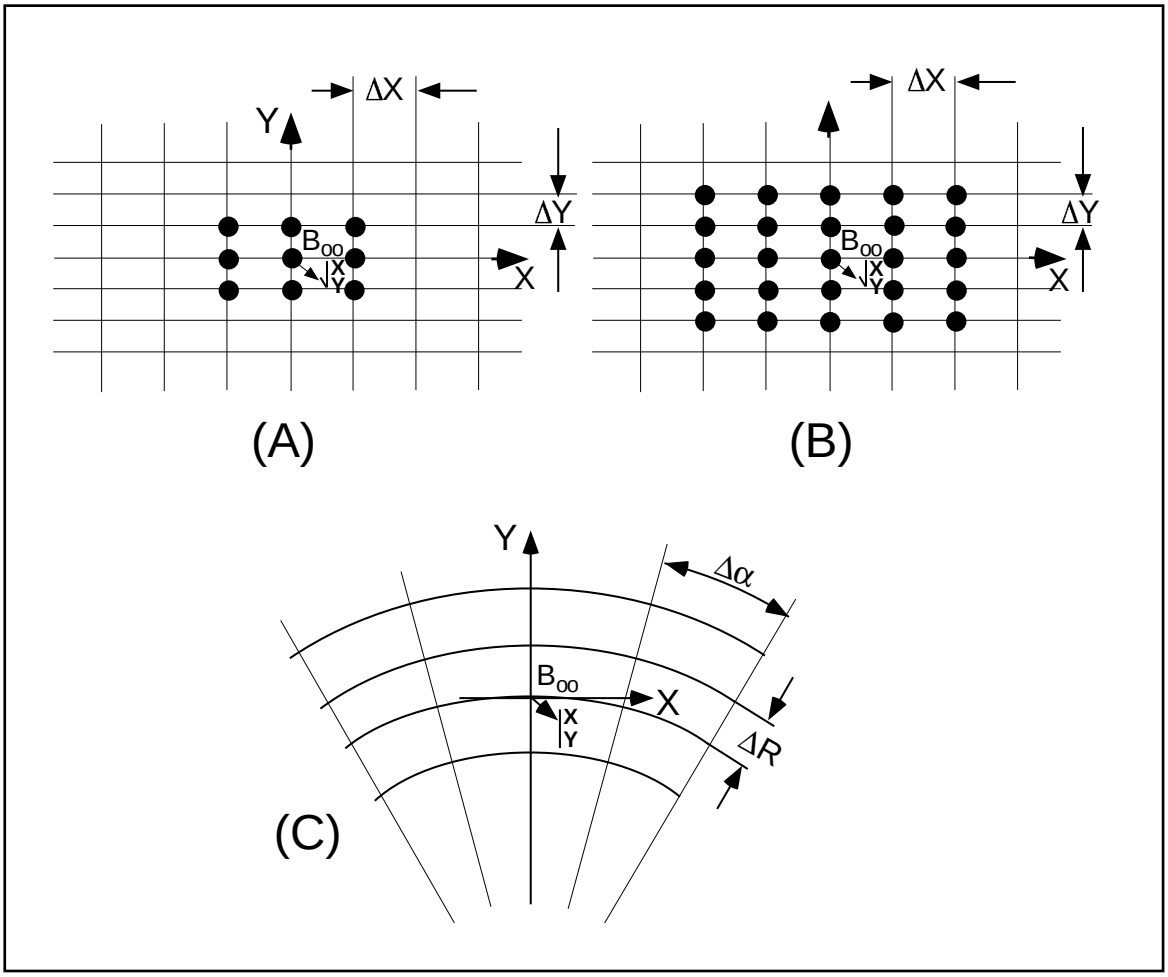


Figure 3: Mesh in the (X,Y) plane. The grid is centered on the node which is closest to the actual position of the particle.
A : Cartesian mesh, 9-point interpolation grid.
B : Cartesian mesh, 25-point interpolation grid.
C : Polar mesh and moving Cartesian frame.

zgoubi provides three types of polynomial interpolation from the mesh (option *IORDRE*) ; namely, a second order interpolation, with either a 9- or a 25-point grid, or a fourth order interpolation with a 25-point grid (Fig. 3).

If the 2-D field map is built up from computer simulation, the grid in principle simply aims at interpolating the field and derivatives at a given point from its 9 or 25 neighbors. On the other hand if the map results from field measurements, the grid also has the virtue of smoothing field fluctuations.

The mesh may be defined in Cartesian coordinates, (Figs. 3A and 3B) or in polar coordinates (Fig. 3C).

The interpolation grid is centered on the node which is closest to the projection in the (X, Y) plane of the actual point of the trajectory.

The interpolation polynomial is

$$B(X, Y, 0) = A_{00} + A_{10}X + A_{01}Y + A_{20}X^2 + A_{11}XY + A_{02}Y^2 \tag{1.4.4}$$

to the second order, or

$$\begin{aligned} B(X, Y, 0) =& A_{00} + A_{10}X + A_{01}Y + A_{20}X^2 + A_{11}XY + A_{02}Y^2 \\ &+ A_{30}X^3 + A_{21}X^2Y + A_{12}XY^2 + A_{03}Y^3 \\ &+ A_{40}X^4 + A_{31}X^3Y + A_{22}X^2Y^2 + A_{13}XY^3 + A_{04}Y^4 \end{aligned} \tag{1.4.5}$$

to the fourth order. The coefficients A_{ij} are calculated by expressions that minimize, with respect to A_{ij}, the quadratic sum

$$S = \sum_{ij} (B(X, Y, 0) - B_{ij})^2 \tag{1.4.6}$$

The source code contains the explicit analytical expressions of the coefficients A_{ij} solutions of the normal equations $\partial S/\partial A_{ij} = 0$.

The A_{ij} may then be identified with the derivatives of $B(X, Y, 0)$ at the central node of the grid

$$A_{ij} = \frac{1}{i!j!} \frac{\partial^{i+j} B}{\partial X^i \partial Y^j} (0, 0, 0) \tag{1.4.7}$$

The derivatives of $B(X, Y, 0)$ with respect to X and Y, at the actual point $(X, Y, 0)$ are obtained by differentiation of the interpolation polynomial, which gives (*e.g.*, from (1.4.4) in the case of second order interpolation)

$$\begin{aligned} \frac{\partial B}{\partial X}(X, Y, 0) &= A_{10} + 2A_{20}X + A_{11}Y \\ \frac{\partial B}{\partial Y}(X, Y, 0) &= A_{01} + A_{11}X + 2A_{02}Y \\ &\text{etc.} \end{aligned} \tag{1.4.8}$$

This allows stepping to the calculation of $\vec{B}(X, Y, Z)$ and its derivatives as described in subsection 1.3.3 (eq. 1.3.3).

The Special Case of Polar Maps

In some optical elements (*e.g., POLARMES, DIPOLE[S]*) the field is given in polar coordinates. It is thus necessary to transform the field and derivatives from the polar frame of the map, (R, α, Z) to the Cartesian moving frame (X, Y, Z), Fig. 3C. This is done as follows.

In second order calculations the correspondence is (we note $B \equiv B_Z(Z = 0)$)

$$
\begin{array}{ll}
\dfrac{\partial B}{\partial X} & = \dfrac{1}{R}\dfrac{\partial B}{\partial \alpha} \\
\dfrac{\partial B}{\partial Y} & = \dfrac{\partial B}{\partial R} \\
\dfrac{\partial^2 B}{\partial X^2} & = \dfrac{1}{R^2}\dfrac{\partial^2 B}{\partial \alpha^2} + \dfrac{1}{R}\dfrac{\partial B}{\partial R} \\
\dfrac{\partial^2 B}{\partial X \partial Y} & = \dfrac{1}{R}\dfrac{\partial^2 B}{\partial \alpha \partial R} - \dfrac{1}{R^2}\dfrac{\partial B}{\partial \alpha} \\
\dfrac{\partial^2 B}{\partial Y^2} & = \dfrac{\partial^2 B}{\partial R^2} \\
\dfrac{\partial^3 B}{\partial X^3} & = \dfrac{3}{R^2}\dfrac{\partial^2 B}{\partial \alpha \partial R} - \dfrac{2}{R^3}\dfrac{\partial B}{\partial \alpha} \\
\dfrac{\partial^3 B}{\partial X^2 \partial Y} & = \dfrac{-2}{R^3}\dfrac{\partial^2 B}{\partial \alpha^2} - \dfrac{1}{R^2}\dfrac{\partial B}{\partial R} + \dfrac{1}{R}\dfrac{\partial^2 B}{\partial R^2} \\
\dfrac{\partial^3 B}{\partial X \partial Y^2} & = \dfrac{2}{R^3}\dfrac{\partial B}{\partial \alpha} - \dfrac{2}{R^2}\dfrac{\partial^2 B}{\partial \alpha \partial R} \\
\dfrac{\partial^3 B}{\partial Y^3} & = 0
\end{array}
\tag{1.4.9}
$$

In fourth order calculations the relations above are pushed to fourth order in $X,\ Y$ whereas

$$
\begin{array}{ll}
\dfrac{\partial^3 B}{\partial X^3} & = \dfrac{1}{R^3}\dfrac{\partial^3 B}{\partial \alpha^3} + \dfrac{3}{R^2}\dfrac{\partial^2 B}{\partial \alpha \partial R} - \dfrac{2}{R^3}\dfrac{\partial B}{\partial \alpha} \\
\dfrac{\partial^3 B}{\partial X^2 \partial Y} & = \dfrac{1}{R^2}\dfrac{\partial^3 B}{\partial \alpha^2 \partial R} - \dfrac{2}{R^3}\dfrac{\partial^2 B}{\partial \alpha^2} - \dfrac{1}{R^2}\dfrac{\partial B}{\partial R} + \dfrac{1}{R}\dfrac{\partial^2 B}{\partial R^2} \\
\dfrac{\partial^3 B}{\partial X \partial Y^2} & = \dfrac{1}{R}\dfrac{\partial^3 B}{\partial \alpha \partial R^2} + \dfrac{2}{R^3}\dfrac{\partial B}{\partial \alpha} - \dfrac{2}{R^2}\dfrac{\partial^2 B}{\partial \alpha \partial R} \\
\dfrac{\partial^3 B}{\partial Y^3} & = \dfrac{\partial^3 B}{\partial R^3} \\
\dfrac{\partial^4 B}{\partial X^4} & = \dfrac{1}{R^4}\dfrac{\partial^4 B}{\partial \alpha^4} - \dfrac{8}{R^4}\dfrac{\partial^2 B}{\partial \alpha^2} + \dfrac{6}{R^3}\dfrac{\partial^3 B}{\partial \alpha^2 \partial R} + \dfrac{3}{R^2}\dfrac{\partial^2 B}{\partial R^2} - \dfrac{3}{R^3}\dfrac{\partial B}{\partial R} \\
\dfrac{\partial^4 B}{\partial X^3 \partial Y} & = \dfrac{1}{R^3}\dfrac{\partial^4 B}{\partial \alpha^3 \partial R} - \dfrac{3}{R^4}\dfrac{\partial^3 B}{\partial \alpha^3} + \dfrac{3}{R^2}\dfrac{\partial^3 B}{\partial \alpha \partial R^2} - \dfrac{8}{R^3}\dfrac{\partial^2 B}{\partial \alpha \partial R} + \dfrac{6}{R^4}\dfrac{\partial B}{\partial \alpha} \\
\dfrac{\partial^4 B}{\partial X^2 2Y^2} & = \dfrac{1}{R^4}\dfrac{\partial^2 B}{\partial \alpha^2} - \dfrac{4}{R^3}\dfrac{\partial^3 B}{\partial \alpha^2 \partial R} - \dfrac{2}{R^2}\dfrac{\partial^2 B}{\partial R^2} + \dfrac{2}{R^3}\dfrac{\partial B}{\partial R} + \dfrac{1}{R^2}\dfrac{\partial^4 B}{\partial \alpha^2 \partial R^2} + \dfrac{1}{R}\dfrac{\partial^3 B}{\partial R^3} \\
\dfrac{\partial^4 B}{\partial X \partial Y^3} & = \dfrac{1}{R}\dfrac{\partial^4 B}{\partial \alpha \partial R^3} - \dfrac{3}{R^2}\dfrac{\partial^3 B}{\partial \alpha \partial R^2} + \dfrac{6}{R^3}\dfrac{\partial^2 B}{\partial \alpha \partial R} - \dfrac{6}{R^4}\dfrac{\partial^4 B}{\partial \alpha^4} \\
\dfrac{\partial^4 B}{\partial Y^4} & = \dfrac{\partial^4 B}{\partial R^4}
\end{array}
\tag{1.4.10}
$$

NOTE : In case a particle goes beyond the limits of the field map, the field and its derivatives are extrapolated using a grid at the border of the map, which is the closest to the actual position of the particle. The flag *IEX* attached to the particle (section 5.6.11, p. 184) is then given the value -1.

1.4.3 Arbitrary 2-D Map, no Symmetry

The map is assumed to describe the field $\vec{B}(B_X, B_Y, B_Z)$ in the (X, Y) plane at elevation Z_0. It provides the components $B_{X,ij}$, $B_{Y,ij}$, $B_{Z,ij}$ at each node (i, j) of a 2-D mesh.

The value of $\vec{B}$ and its derivatives at the projection (X, Y, Z_0) of the actual position (X, Y, Z) of a particle is obtained by means of (parameter *IORDRE* in keyword data list - see for instance *MAP2D, MAP2D-E*) either a second degree polynomial interpolation from a 3×3 points grid (*IORDRE*=2), or a fourth degree polynomial interpolation from a 5×5 points grid (*IORDRE*=4), centered at the node (i, j) closest to the position (X, Y).

To second order for instance

$$B_\ell(X,Y,Z_0) = A_{00} + A_{10}X + A_{01}Y + A_{20}X^2 + A_{11}XY + A_{02}Y^2 \tag{1.4.11}$$

where B_ℓ stands for any of the three components B_X, B_Y or B_Z. Differentiating then gives the derivatives

$$\begin{aligned} \frac{\partial B_\ell}{\partial X}(X,Y,Z_0) &= A_{10} + 2A_{20}X + A_{11}Y \\ \frac{\partial^2 B_\ell}{\partial X \partial Y}(X,Y,Z_0) &= A_{11} \\ &\text{etc.} \end{aligned} \tag{1.4.12}$$

Then follows a procedure of extrapolation from (X,Y,Z_0) to the actual position (X,Y,Z), based on Taylor series development.
No special symmetry is assumed, which allows the treatment of arbitrary field distribution (*e.g.*, solenoid, helical snake).

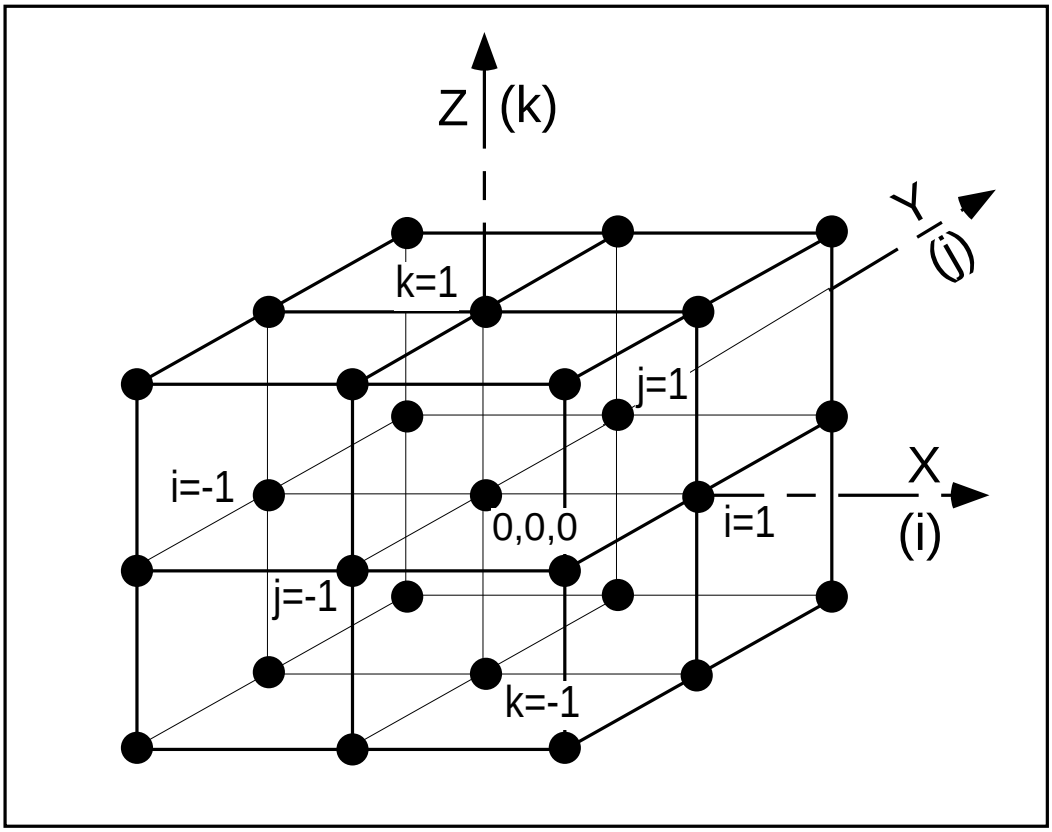


Figure 4: A 3-D 27-point grid is used for interpolation of magnetic or electric fields and derivatives up to second order. The central node of the grid $(i = j = k = 0)$ is the closest to the actual position of the particle.

1.4.4 3-D Field Map

When using a 3-D field map, the vector field $\vec{B}(X,Y,Z)$ and its derivatives necessary for the calculation of position and velocity of the particle are obtained via second degree polynomial interpolation,

$$B_\ell(X,Y,Z) = A_{000} + A_{100}X + A_{010}Y + A_{001}Z + A_{200}X^2 + A_{020}Y^2 + A_{002}Z^2 + A_{110}XY + A_{101}XZ + A_{011}YZ \tag{1.4.13}$$

B_ℓ stands for any of the three components, B_X, B_Y or B_Z. By differentiation of B_ℓ one gets

$$\begin{aligned} \frac{\partial B_\ell}{\partial X} &= A_{100} + 2A_{200}X + A_{110}Y + A_{101}Z \\ \frac{\partial^2 B_\ell}{\partial X^2} &= 2A_{200} \end{aligned} \tag{1.4.14}$$

and so on up to second order derivatives with respect to X, Y or Z.
The interpolation involves a $3 \times 3 \times 3$-point parallelepipedic grid (Fig. 4), the origin of which is positioned at the node of the 3-D field map which is closest to the actual position of the particle.

Let B^ℓ_{ijk} be the value of the — measured or computed — magnetic field at each one of the 27 nodes of the 3-D grid (B^ℓ stands for B_X, B_Y or B_Z), and $B_\ell(X, Y, Z)$ be the value at a position (X, Y, Z) with respect to the central node of the 3-D grid. Thus, any coefficient A_i of the polynomial expansion of B_ℓ is obtained by means of expressions that minimize, with respect to A_i, the sum

$$S = \sum_{ijk} \left(B_\ell(X, Y, Z) - B^\ell_{ijk} \right)^2 \tag{1.4.15}$$

where the indices i, j and k take the values -1, 0 or +1 so as to sweep the 3-D grid. The source code contains the explicit analytical expressions of the coefficients A_{ijk} solutions of the normal equations $\partial S / \partial A_{ijk} = 0$.

2 SPIN TRACKING

Spin motion tracking was implemented in **zgoubi** in the SATURNE years [10], it has been since used in a number of studies, including the transport of polarized muons in the Neutrino Factory [11], polarization survival in the superB project [12], studies on proton polarization with helical snakes at AGS [13] and RHIC [14], and in the magic-energy electrostatic ring of the proton-EDM project [15].

2.1 Introduction

The precession of the spin $\vec{S}$ of a charged particle in electric and magnetic fields is governed by the Thomas-BMT first order differential equation [16]

$$\frac{d\vec{S}}{dt} = \frac{q}{m}\,\vec{S}\times\vec{\omega} \tag{2.1.1}$$

with, in the laboratory frame,

$$\vec{\omega} = (1+\gamma G)\vec{b} + G(1-\gamma)\vec{b}_{/\!/} + \gamma(G+\frac{1}{1+\gamma})\frac{\vec{e}\times\vec{v}}{c^2} \tag{2.1.2}$$

t, q, m, c, γ and G are respectively the time, charge, relativistic mass, speed of light, Lorentz relativistic factor and anomalous magnetic moment of the particle, $\vec{b}$ and $\vec{e}$ are the fields in the laboratory, $\vec{b}_{/\!/}$ is the component of $\vec{b}$ parallel to the velocity $\vec{v}$ of the particle.
Equation (2.1.1) is normalized by introducing the same notation as for the Lorentz equation, page 19, namely $b = \| \vec{b} \|$, $v = ||\vec{v}||$, $\vec{v} = v\vec{u}$, $ds = vdt$ the differential path, $\frac{\gamma m v}{q} = B\rho$ the rigidity of the particle, whereas $\vec{S}' = \frac{d\vec{S}}{ds} = \frac{1}{v}\frac{d\vec{S}}{dt}$ is the derivative of the spin with respect to the path. This yields

$$(B\rho)\,\vec{S}' = \vec{S}\times\vec{\omega} \quad \text{or} \quad \vec{S}' = \vec{S}\times\vec{\Omega} \tag{2.1.3}$$

where, noting $\vec{B} = \vec{b}/B\rho$, $\vec{E} = \vec{e}/B\rho$,

$$\vec{\Omega} = \frac{\vec{\omega}}{B\rho} = (1+\gamma G)\vec{B} + G(1-\gamma)\vec{B}_{/\!/} + \frac{\beta\gamma}{c}(G+\frac{1}{1+\gamma})\vec{E}\times\vec{u} \tag{2.1.4}$$

This equation is solved in the same way as the reduced Lorentz equation (1.2.3). From the values of the precession factor $\vec{\omega}(M_0)$ and spin $\vec{S}(M_0)$ of the particle at position M_0, the spin $\vec{S}(M_1)$ at position M_1, following a displacement Δs (fig. 2), is obtained from truncated Taylor expansion

$$\vec{S}(M_1) \approx \vec{S}(M_0) + \frac{d\vec{S}}{ds}(M_0)\,\Delta s + \frac{d^2\vec{S}}{ds^2}(M_0)\frac{\Delta s^2}{2} + \frac{d^3\vec{S}}{ds^3}(M_0)\frac{\Delta s^3}{3!} + \frac{d^4\vec{S}}{ds^4}(M_0)\frac{\Delta s^4}{4!} + \frac{d^5\vec{S}}{ds^5}(M_0)\frac{\Delta s^5}{5!} \tag{2.1.5}$$

2.2 Integration in Magnetic Fields

In purely magnetic fields $\vec{e} = 0$ thus eq. (2.1.4) reduces to

$$\vec{\Omega}_b = (1+\gamma G)\vec{B} + G(1-\gamma)\vec{B}_{/\!/} \tag{2.2.1}$$

$\vec{S}$ and its derivatives $\vec{S}^{(n)} = d^n\vec{S}/ds^n$ satisfy the recursive differentiation relations

$$\begin{aligned}
\vec{S}' &= \vec{S} \times \vec{\Omega}_b \\
\vec{S}'' &= \vec{S}' \times \vec{\Omega}_b + \vec{S} \times \vec{\Omega}'_b \\
\vec{S}''' &= \vec{S}'' \times \vec{\Omega}_b + 2\vec{S}' \times \vec{\Omega}'_b + \vec{S} \times \vec{\Omega}''_b \\
&\text{etc.}
\end{aligned} \tag{2.2.2}$$

with the derivatives $d^n\vec{\Omega}_b/ds^n$ obtained by differentiation of eq. (2.2.1). This requires $\vec{B}_{/\!/}$ and derivatives, obtained in the following way,

$$\begin{aligned}
\vec{B}_{/\!/} &= (\vec{B} \cdot \vec{u})\,\vec{u} \\
\vec{B}'_{/\!/} &= (\vec{B}' \cdot \vec{u} + \vec{B} \cdot \vec{u}')\,\vec{u} + (\vec{B} \cdot \vec{u})\,\vec{u}' \\
\vec{B}''_{/\!/} &= (\vec{B}'' \cdot \vec{u} + 2\vec{B}' \cdot \vec{u}' + \vec{B} \cdot \vec{u}'')\,\vec{u} + 2(\vec{B}' \cdot \vec{u} + \vec{B} \cdot \vec{u}')\,\vec{u}' + (\vec{B} \cdot \vec{u})\,\vec{u}'' \\
&\text{etc.}
\end{aligned} \tag{2.2.3}$$

The quantities $\vec{u}$, $\vec{B}$ and their derivatives as involved in these equations are known, being sub-products of the integration of the motion of the particle, eqs. (1.2.8, 1.2.10) p. 21.

2.3 Integration in Electric Fields

In purely electric fields $\vec{b} = 0$, thus eq. (2.1.2) reduces to

$$\omega_e = \frac{\beta\gamma}{c}(G + \frac{1}{1+\gamma})\,\vec{e} \times \vec{u} \tag{2.3.1}$$

$\vec{S}$ and its derivatives $\vec{S}^{(n)} = d^n\vec{S}/ds^n$ satisfy the recursive differentiation relations

$$\begin{aligned}
B\rho\,\vec{S}' &= \vec{S} \times \vec{\omega}_e \\
(B\rho)'\vec{S}' + B\rho\vec{S}'' &= \vec{S}' \times \vec{\omega}_e + \vec{S} \times \vec{\omega}'_e \\
(B\rho)''\vec{S}' + 2(B\rho)'\vec{S}'' + B\rho\vec{S}''' &= \vec{S}'' \times \vec{\omega}_e + 2\vec{S}' \times \vec{\omega}'_e + \vec{S}' \times \vec{\omega}''_e \\
&\text{etc.}
\end{aligned} \tag{2.3.2}$$

that provide the derivatives $d^n\vec{S}/ds^n$ needed in the Taylor expansion (eq. 2.1.5). The derivatives $(B\rho)^{(n)} = d^n(B\rho)/ds^n$ above are a sub-product of the integration of the force law (eq. 1.2.18). The derivatives of ω_e (eq. 2.3.1) are obtained by recursive differentiation,

$$\begin{aligned}
\vec{\omega}'_e &= \left(\frac{\beta\gamma}{c}(G + \frac{1}{1+\gamma})\right)' \vec{e} \times \vec{u} + \frac{\beta\gamma}{c}(G + \frac{1}{1+\gamma})\,(\vec{e} \times \vec{u})' \\
\vec{\omega}''_e &= \left(\frac{\beta\gamma}{c}(G + \frac{1}{1+\gamma})\right)'' \vec{e} \times \vec{u} + 2\left(\frac{\beta\gamma}{c}(G + \frac{1}{1+\gamma})\right)' (\vec{e} \times \vec{u})' + \frac{\beta\gamma}{c}(G + \frac{1}{1+\gamma})\,(\vec{e} \times \vec{u})'' \\
&\text{etc.}
\end{aligned} \tag{2.3.3}$$

The quantities $\left(\frac{1}{1+\gamma}\right)^{(n)}$ in the $(\omega_e)^{(n)}$ above are obtained as follows.

From $\gamma^2 = 1/(1-\beta^2)$ it comes $(\gamma+1)(\gamma-1)\beta^2/(1-\beta^2)$ and then

$$\frac{1}{1+\gamma} = \left(\frac{1}{\beta}-1\right)\left(\frac{1}{\beta}+1\right)(\gamma-1)$$

which is easily differentiated, recursively,

$$\begin{aligned}
\left(\frac{1}{1+\gamma}\right)' &= \left(\frac{1}{\beta}\right)'\left(\frac{1}{\beta}+1\right)(\gamma-1) + \left(\frac{1}{\beta}-1\right)\left(\frac{1}{\beta}\right)'(\gamma-1) + \left(\frac{1}{\beta}-1\right)\left(\frac{1}{\beta}+1\right)\gamma' \\
\left(\frac{1}{1+\gamma}\right)'' &= \left(\frac{1}{\beta}\right)''\left(\frac{1}{\beta}+1\right)(\gamma-1) + \left(\frac{1}{\beta}-1\right)\left(\frac{1}{\beta}\right)''(\gamma-1) + \left(\frac{1}{\beta}-1\right)\left(\frac{1}{\beta}+1\right)\gamma'' + \\
&\quad 2\left(\frac{1}{\beta}\right)'\left(\frac{1}{\beta}\right)(\gamma-1) + 2\left(\frac{1}{\beta}\right)'\left(\frac{1}{\beta}\right)'(\gamma-1) + 2\left(\frac{1}{\beta}-1\right)\left(\frac{1}{\beta}\right)'\gamma' \\
&\text{etc.}
\end{aligned} \tag{2.3.4}$$

The interest of that formulation is that the $\left(\frac{1}{\beta}\right)^{(n)}$ are already known from the particle dynamics, eq. (1.2.21), as well as the derivatives

$$\frac{d^{n+1}\gamma}{ds^{n+1}} = \frac{q}{m}\frac{d^n(\vec{e}\cdot\vec{u})}{ds^n}$$

following eq. (1.2.20), whereas the $d^n(\vec{e}\cdot\vec{u})/ds^n$ are given by eq. (1.2.19).

2.4 Integration in Combined Electric and Magnetic Fields

When both $\vec{e}$ and $\vec{b}$ are non-zero, the complete eqs. (2.1.3, 2.1.4) must be considered.

The precession vector (eq. 2.1.4) and its derivatives can be split into independent, magnetic and electric components, namely

$$\vec{\omega} = \vec{\omega}_b + \vec{\omega}_e, \quad \vec{\omega}' = \vec{\omega}_b' + \vec{\omega}_e', \qquad \text{etc.}$$

As a consequence, the spin vector $\vec{S}(s)$ and its derivatives at location M_0 prior to a Δs push to location M_1 (fig. 2) can be obtained by linear superposition of the separate solutions for the magnetic case (eq. 2.2.2) and for the electric case (eq. 2.3.2), namely

$$
\begin{aligned}
&B\rho\,\vec{S}' = \\
&\quad \vec{S}\times\vec{\omega}_b && +\vec{S}\times\vec{\omega}_e \\
&B\rho\vec{S}'' = \\
&\quad \vec{S}'\times\vec{\omega}_b + \vec{S}\times\vec{\omega}_b' && +\vec{S}'\times\vec{\omega}_e + \vec{S}\times\vec{\omega}_e' - (B\rho)'\vec{S}' \\
&B\rho\vec{S}''' = \\
&\quad \underbrace{\vec{S}''\times\vec{\omega}_b + 2\vec{S}'\times\vec{\omega}_b' + \vec{S}\times\vec{\omega}_b''}_{\text{Magnetic field component, eq. (2.2.2)}} && \underbrace{+\vec{S}''\times\vec{\omega}_e + 2\vec{S}'\times\vec{\omega}_e' + \vec{S}\times\vec{\omega}_e'' - \Big((B\rho)''\vec{S}' + 2(B\rho)'\vec{S}''\Big)}_{\text{Electric field component, eq. (2.3.2)}} \\
&\quad \text{etc.}
\end{aligned}
\tag{2.4.1}
$$

The process is then completed by applying the Δs push, eq. (2.1.5).

2.5 Radiative polarization

Radiative spin transition is accounted for in a classical manner with an additional term in the Thomas-BMT differential equation for the motion of the polarization vector [17], namely, see Eqs. 2.1.1, 2.1.2 (in the absence of electric fields),

$$\frac{d\vec{S}}{dt} = \frac{q}{m}\left(\vec{S}\times\vec{(}1+a\gamma)\vec{b} + a(1-\gamma)\vec{b}_{/\!/}\right) - \frac{1}{\tau_{ST}}\left(\vec{S} - \frac{2}{9}(\vec{S}\cdot\vec{u})\vec{u} + \frac{8}{5\sqrt{3}}\frac{\vec{u}\times\dot{\vec{u}}}{|\dot{\vec{u}}|}\right) \tag{2.5.1}$$

where the electron notation for the magnetic anomaly, a, replaces hadron's G (and $q=-e$ with e the elementary charge, for electrons), $\vec{u}$ is the unit velocity vector of the particle and $\dot{\vec{u}} = d\vec{u}/dt = v\vec{u}'$ (eq. 1.2.2). The Sokolov-Ternov polarization equilibrium time constant has been introduced,

$$\tau_{ST} = \frac{4}{5\sqrt{3}}\frac{E}{\hbar\omega_c}\tau_0 \tag{2.5.2}$$

with $E=\gamma m_0$ the particle energy and $\hbar\omega_c = h\times 3\gamma^3 c\,/\,2\rho$ the critical photon energy.

This radiative term is only added if the particle does radiate [18], that event being determined by the Monte Carlo photon emission simulation process described in Sec. 3.1. It has the effect of slightly changing the electron polarization, under the effect of radiation induced spin transition.

This radiative correction process is under installation in **zgoubi**. Its benchmarking includes comparison with statistical averages of quantum spin physics.

As to solving the differential eq. 2.5.3, the methods follows the **zgoubi** method, namely, introducing derivatives with respect to the path variable s, and fields normalized to $B\rho$,

$$\frac{d\vec{S}}{ds} = \left(\vec{S}\times\vec{(}1+a\gamma)\vec{B} + a(1-\gamma)\vec{B}_{/\!/}\right) - \frac{1}{\tau_{ST}}\left(\vec{S} - \frac{2}{9}(\vec{S}\cdot\vec{\beta})\vec{\beta} + \frac{8}{5\sqrt{3}}\frac{\vec{\beta}\times\dot{\vec{\beta}}}{|\dot{\vec{\beta}}|}\right) \tag{2.5.3}$$

The derivatives of the unit velocity vector $\vec{u}$ are outcomes of the solution of particle motion as detailed in Sec. 1.2. The derivatives of the polarization vector are computed as detailed in ec. 2.1.

2.6 Spin Diffusion

To be documented.

3 SYNCHROTRON RADIATION

zgoubi provides the simulation of two distinct types of synchrotron radiation (SR) manifestations namely, on the one hand energy loss by stochastic emission of photons and the ensuing perturbation on particle dynamics and, on the other hand the radiated electro-magnetic field impulse and its spectral-angular energy density as observed in the laboratory.

SR loss simulation was first installed in **zgoubi** in view of beam dynamics studies in the beam delivery system of the Next Linear Collider [19], based on a method developed in the frame of the ELFE project ("EU Lab. For Electrons") [20, 21, 22]. It was next used for including damping effects in beam studies regarding various electron-ring and recirculator projects [12, 23].

SR electromagneitc impulse and spectrum computation was first installed in **zgoubi** for the study of interference effects at the LEP beam diagnostics miniwiggler [25]. These simulation tools were next used to design the LHC SR beam diagnostics systems, located in the IR4 RF section [26].

3.1 Energy Loss and Dynamical Effects

Most of the contents of the present section are drawn from Ref. [19].

Given a particle wandering in the magnetic field of an arbitrary optical element or field map, **zgoubi** computes the energy loss undergone, and its effect on the particle motion. The energy loss is calculated in a classical manner, by invoking two random processes that accompany the emission of a photon namely,

- the probability of emission,
- the energy of the photon.

The effects on the dynamics of the emitting particle either account for the sole alteration of its energy, or, if requested, include the angle kick. Particle position is supposed not to change upon emission of a photon. These calculations and ensuing dynamics corrections are performed at each integration step. In a practical manner, this requires centimeters to tens of centimeters steps in smoothly varying magnetic fields (a quantity to be determined before any simulation, from convergence trials).

Main aspects of the method are developed in the following.

Probability of Emission of a Photon

Given that the number of photons emitted within a step Δs can be very small (units or fractions of a unit)[1] a Poisson probability law

$$p(k) = \frac{\lambda^k}{k!} \exp(-\lambda) \tag{3.1.1}$$

is considered. k is the number of photons emitted over a $\Delta\theta$ (circular) arc of trajectory such that, the average number of photons per radian expresses as[2]

[1] For instance, a 1 GeV electron will emit about 20.6 photons per radian ; an integration step size $\Delta s = 0.1$ m upon $\rho = 10$ m bending radius results in 0.2 photon per step.

[2] This leads for instance, in the case of electrons, to the classical formula $\lambda/\Delta\theta \approx 129.5\text{E(GeV)}/2\pi \approx \gamma/94.9$.

$$\lambda = \frac{20 e r_0}{8\hbar\sqrt{3}} \beta^2 B\rho \frac{\Delta s}{\rho} \tag{3.1.2}$$

where $r_0 = e^2/4\pi\epsilon_0 m_0 c^2$ is the classical radius of the particle of rest-mass m_0, e is the elementary charge, $\hbar = h/2\pi$, h is the Planck constant, $\beta = v/c$, $B\rho$ is the particle stiffness. λ is evaluated at each integration step from the current values β, $B\rho$ and Δs, then a value of k is drawn by a rejection method.

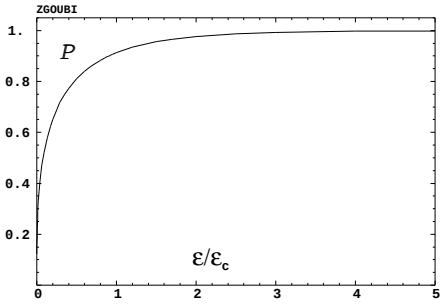
Energy of the Photons

These k photons are assigned energies $\epsilon = h\nu$ at random, in the following way. The cumulative distribution of the energy probability law $p(\epsilon/\epsilon_c)d\epsilon/\epsilon_c$ (*i.e.*, the probability of emission of a photon with energy in $[0, \epsilon]$) writes [3]

$$\mathcal{P}(\epsilon/\epsilon_c) = \frac{3}{5\pi} \int_0^{\epsilon/\epsilon_c} \frac{d\epsilon}{\epsilon_c} \int_{\epsilon/\epsilon_c}^{\infty} K_{5/3}(x) dx \tag{3.1.3}$$

where $K_5/3$ is a modified Bessel function and, $\epsilon_c = \hbar\omega_c$, $\omega_c = 2\pi \times 3\gamma^3 c/2\rho$ being the critical frequency in the presence of bending radius ρ ; ω_c is evaluated at each integration step from the current values γ and ρ, in other words, this energy loss calculation assumes constant magnetic field over the trajectory arc Δs. In the low frequency region $(\epsilon/\epsilon_c \ll 1)$ it can be approximated by

$$\mathcal{P}(\epsilon/\epsilon_c) = \frac{12\sqrt{3}}{2^{1/3}\, 5\, \Gamma(\frac{1}{3})} (\frac{\epsilon}{\epsilon_c})^{1/3} \tag{3.1.4}$$



Cumulative energy distribution $\mathcal{P}(\epsilon/\epsilon_c)$.

About 40 values of $\mathcal{P}(\epsilon/\epsilon_c)$ computed from eq. 3.1.3 [24], honestly spread over a range $\epsilon/\epsilon_c \leq 10$ are tabulated in **zgoubi** source file (see figure). In order to get ϵ/ϵ_c, first a random value $0 < \mathcal{P} < 1$ is generated uniformly, then ϵ/ϵ_c is drawn either by simple inverse linear interpolation of the tabulated values if $\mathcal{P} > 0.26$ (corresponding to $\epsilon/\epsilon_c > 10^{-2}$), or, if $\mathcal{P} < 0.26$ from eq. 3.1.4 that gives $\epsilon/\epsilon_c = \big(\frac{5\, 2^{1/3}\Gamma(\frac{1}{3})}{12\sqrt{3}\mathcal{P}}\big)^3$ with precision no less than 1% at $\mathcal{P} \to 0.26$.

When SR loss tracking is requested, several optical elements that contain a dipole magnetic field component (*e.g.*, *MULTIPOL*, *BEND*) provide a printout of various quantities related to SR emission, as drawn from classical theoretical expressions, such as for instance,
- energy loss per particle $\Delta E(eV) = \frac{2}{3} r_0 c \gamma^3 B(T) \Delta\theta$, with B the dipole field exclusive of any other multipole component in the magnet, $\Delta\theta$ the total deviation as calculated from B, from the magnet length, and from the reference rigidity *BORO* (as defined with, *e.g.*, *[MC]OBJET*)
- critical energy $\epsilon_c(eV) = \frac{3\gamma^3 c}{2\rho} \frac{\hbar}{e}$, with ρ =*BORO*$/B$
- average energy of the photons radiated $< \epsilon >= \frac{8}{15\sqrt{3}} \epsilon_c$,
- *rms* energy of radiated photons $\epsilon_{rms} = 0.5591 \epsilon_c$,
- number of radiated photons per particle $N = \Delta E / < \epsilon >$.

This is done in order to facilitate verifications, since on the other hand statistics regarding those values are drawn from the tracking and may be printed using the dedicated keyword *SRPRNT*.

[3] From a practical viewpoint, the value of the magnetic field first computed for a one-step push of the particle (eqs. 1.2.4, 1.2.8) is next used to obtain ρ and perform SR loss corrections afterwards.

Finally, upon user's request as well, SR loss can be limited to particular classes of optical elements, for instance dipole magnets alone, or dipole + quadrupole magnets, etc. This option is made available in order to allow further inspection, or easier comparison with other codes, for instance.

3.2 Spectral-Angular Radiated Densities

Most of the content in the present section is drawn from Refs. [25, 26, 27].

The ray-tracing procedures provide the ingredients necessary for the determination of the electric field radiated by the particle subject to acceleration, as shown in Fig. 5, this is developed in section 3.2.1. These ingredients further allow calculating the spectral-angular density of the radiation[4], this is developed in section 3.2.2.

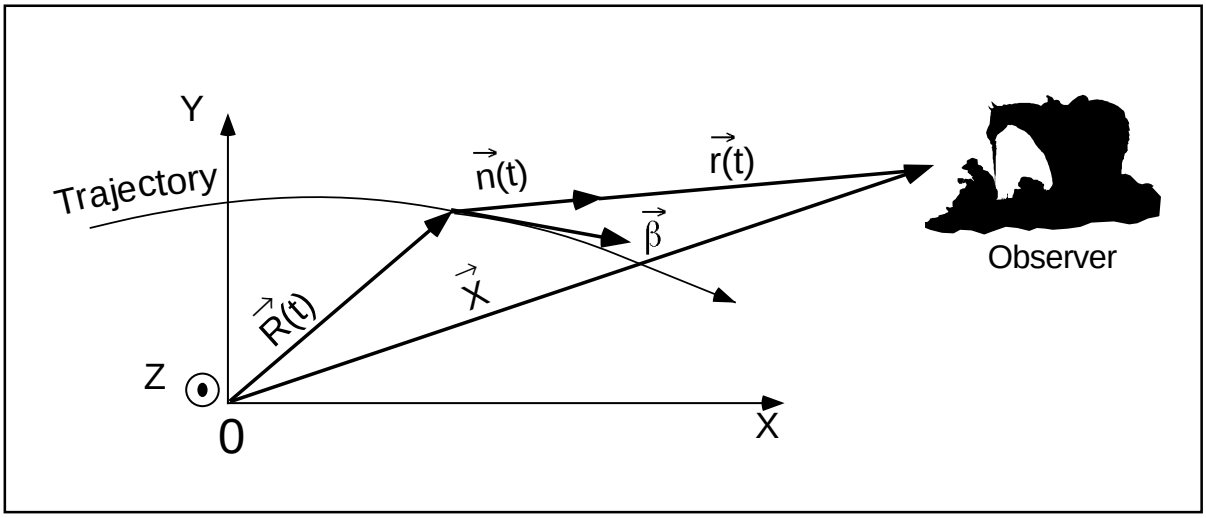


Figure 5: A scheme of the reference frame in **zgoubi** together with the vectors entering in the definition of the electric field radiated by the accelerated particle :
(x, y) : horizontal plane ; z : vertical axis.
$\vec{R}(t)$ = particle position in the fixed frame (O, x, y, z) ;
$\vec{X}$ (time-independent) = position of the observer in the (O, x, y, z) frame ;
$\vec{r}(t) = \vec{X} - \vec{R}(t)$ = position of the particle with respect to the observer ;
$\vec{n}(t)$ = (normalized) direction of observation = $\vec{r}(t)/|\vec{r}(t)|$;
$\vec{\beta}$ = normalized velocity vector of the particle $\vec{v}/c = (1/c)d\vec{R}/dt$.

3.2.1 Calculation of the Radiated Electric Field

The expression for the radiated electric field $\vec{\mathcal{E}}(\vec{n}, \tau)$ as seen by the observer in the long distance approximation is [28]

$$\vec{\mathcal{E}}(\vec{n}, \tau) = \frac{q}{4\pi\varepsilon_0 c} \frac{\vec{n}(t) \times \left[\left(\vec{n}(t) - \vec{\beta}(t) \right) \times d\vec{\beta}/dt \right]}{r(t) \left(1 - \vec{n}(t) \cdot \vec{\beta}(t) \right)^3} \tag{3.2.1}$$

where t is the time in which the particle motion is described and τ is the observer time. Namely, when at position $\vec{r}(t)$ with respect to the observer (or as well at position $\vec{R}(t) = \vec{X} - \vec{r}(t)$ in the (O, x, y, z) frame) the particle emits a signal which reaches the observer at time τ, such that $\tau = t + r(t)/c$ where $r(t)/c$ is the delay necessary for the signal to travel from the emission point to the observer, which also leads by differentiation to the well-known relation

[4] These calculations have been installed in the post-processor **zpop**.

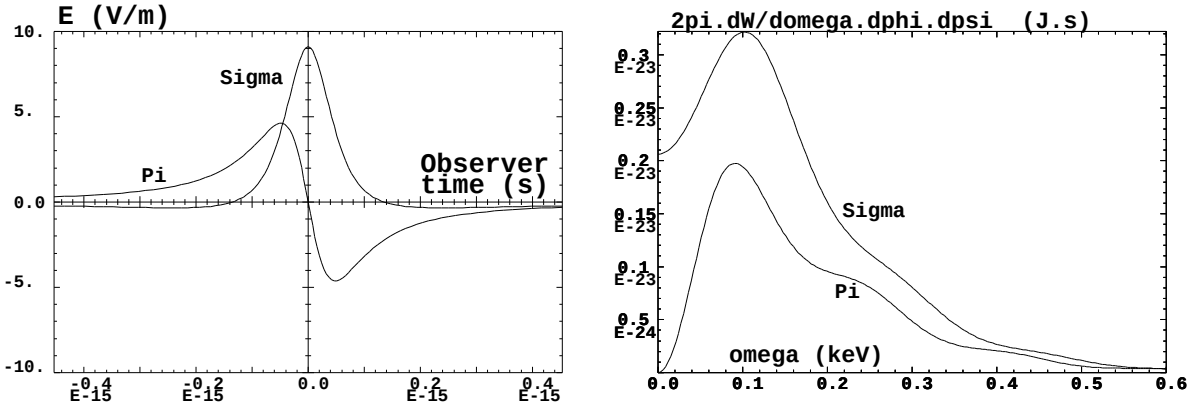


Figure 6: *Left* : typical shapes of the $\mathcal{E}_\sigma(t)$ and $\mathcal{E}_\pi(t)$ components of the electric field impulse (eq. 3.2.1) emitted by a 2.5 GeV electron on a $\rho = 53.6$ m circular trajectory in a $l = 20$ cm-long dipole, as observed in the direction of the centre of the dipole, $\phi = 0$, $\gamma\psi = 5$ (observation distance $r = 10$ m). *Right* : the related spectral angular densities $\partial^3 W_{\sigma,\pi}/\partial\phi\,\partial\psi\,\partial\omega$ (eq. 3.2.10). After Ref. [27].

$$\frac{d\tau}{dt} = 1 - \vec{n}(t)\cdot\vec{\beta}(t) \tag{3.2.2}$$

The vectors $\vec{R}(t)$ and $\vec{\beta}(t) = \dfrac{v}{c}\vec{u}$ (eq. 1.2.2) that describe the motion are obtained from the ray-tracing (eqs. 1.2.4). The acceleration is calculated from (a form of eq. 1.2.1 with $\vec{e} = 0$)

$$\frac{d\vec{\beta}}{dt} = \frac{q}{m}\,\vec{\beta}(t)\times\vec{b}(t) \tag{3.2.3}$$

Then, given the observer position $\vec{X}$ in the fixed frame, it is possible to calculate

$$\vec{r}(t) = \vec{X} - \vec{R}(t) \quad \text{and} \quad \vec{n}(t) = \frac{\vec{r}(t)}{|\vec{r}(t)|} \tag{3.2.4}$$

As an illustration, Fig. 6 has been produced using **zpop**, it shows the typical shape of the electric field impulse at the observer, with central peak width [27]

$$2\,\tau_c = 2\,\frac{2\rho}{3\gamma^3\,c}\,(1+\gamma^2\psi^2)^{3/2}$$

Computation of $\vec{n}-\vec{\beta}$ and $1-\vec{n}\cdot\vec{\beta}$

Owing to computer precision the crude computation of $\vec{n}-\vec{\beta}$ and $1-\vec{n}\cdot\vec{\beta}$ may lead to

$$\vec{n}-\vec{\beta} = 0 \text{ and } 1-\vec{n}\cdot\vec{\beta} = 0$$

since the preferred direction of observation is generally almost parallel to $\vec{\beta}$ (in that case, parallel in the sense of computer precision), while $\beta \approx 1$ as soon as particle energies of a few hundred times the rest mass are concerned.
It is therefore necessary to express $\vec{n}-\vec{\beta}$ and $1-\vec{n}\cdot\vec{\beta}$ in an adequate form for achieving accurate software computation.
The expression for $\vec{n}$ is

$$\begin{aligned}\vec{n} = (n_x, n_y, n_z) &= (\cos\psi\,\cos\phi, \cos\psi\,\sin\phi, \sin\psi)\\ &= \left[1 - 2(\sin^2\phi/2 + \sin^2\psi/2) + 4\sin^2\phi/2\sin^2\psi/2, \sin\phi(1-2\sin^2\psi/2), \sin\psi\right]\end{aligned} \tag{3.2.5}$$

where ϕ and ψ are the observation angles, given by

$$\phi = \text{Atg}\left(\frac{r_y}{r_x}\right) \text{ and } \psi = \text{Atg}\left(\frac{r_z}{\sqrt{r_x^2 + r_y^2}}\right) \tag{3.2.6}$$

with $\vec{r} = (r_x, r_y, r_z)$, while $\vec{\beta}$ can be written under the form

$$\begin{aligned}\vec{\beta} = (\beta_x, \beta_y, \beta_z) &= \left[\sqrt{(\beta^2 - \beta_y^2 - \beta_z^2)}, \beta_y, \beta_z\right] \\ &= \left[\sqrt{(1 - 1/\gamma^2 - \beta_y^2 - \beta_z^2)}, \beta_y, \beta_z\right] = (1 - a/2 + a^2/8 - a^3/16 + ..., \beta_y, \beta_z)\end{aligned} \tag{3.2.7}$$

where $a = 1/\gamma^2 + \beta_y^2 + \beta_z^2$. This leads to

$$n_x = 1 - \varepsilon_x \text{ and } \beta_x = 1 - \xi_x$$

with

$$\varepsilon_x = 2(\sin^2 \phi/2 + \sin^2 \psi/2) - 4 \sin^2 \phi/2 \sin^2 \psi/2$$

and

$$\xi_x = a/2 - a^2/8 + a^3/16 + ...$$

All this provides, on the one hand,

$$\vec{n} - \vec{\beta} = (-\varepsilon_x + \xi_x, n_y - \beta_y, n_z - \beta_z)\ , \tag{3.2.8}$$

whose components are combinations of terms of the same order of magnitude (ε_x and $\xi_x \sim 1/\gamma^2$ while n_y, β_y, n_z and $\beta_z \sim 1/\gamma$) and, on the other hand,

$$1 - \vec{n} \cdot \vec{\beta} = \varepsilon_x + \xi_x - n_y\beta_y - n_z\beta_z - \varepsilon_x\xi_x\ , \tag{3.2.9}$$

that combines terms of the same order of magnitude ($\varepsilon_x, \xi_x, n_y\beta_y$ and $n_z\beta_z \sim 1/\gamma^2$), plus $\varepsilon_x\beta_x \sim 1/\gamma^4$. The precision of these expressions is directly related to the order at which the series

$$\xi_x = a/2 - a^2/8 + a^3/16 + ... \qquad (a = 1/\gamma^2 + \beta_y^2 + \beta_z^2)$$

is pushed, however the convergence is fast since $a \sim 1/\gamma^2 \ll 1$ in situations of concern.

3.2.2 Calculation of the Fourier Transform of the Electric Field

The Fourier transforms

$$FT_\omega[\vec{\mathcal{E}}(\tau)] = \int \vec{\mathcal{E}}(\tau) e^{-i\omega\tau} d\tau$$

of the σ and π electric field components provide the spectral angular energy density

$$\frac{\partial^3 W}{\partial\phi\, \partial\psi\, \partial\omega} = \frac{r^2}{\mu_0 c} \left| FT_\omega \left(\vec{\mathcal{E}}(\tau)\right)\right|^2 \tag{3.2.10}$$

Fig. 6-right gives a typical example in the case of a short magnet. These Fourier transforms are computed without resorting to FFT techniques, namely from

$$FT_\omega \left[\vec{\mathcal{E}}(\tau)\right] \approx \sum \vec{\mathcal{E}}(\tau_k)\, e^{-i\omega\tau_k}\, \Delta\tau_k \tag{3.2.11}$$

for two reasons. On the one hand, the number of integration steps Δs that define the trajectory (eqs. 1.2.4), is arbitrary and therefore in general not of order 2^n. On the other hand, the integration step defines a constant time differential element $\Delta t_k = \Delta s/\beta c$ which results in the observer differential time element $\Delta\tau_k$, which is also the differential element of the Fourier transform, being non-constant, since both are related by eq. 3.2.2 in which $\vec{\beta}$ and $\vec{n}$ vary as a function of the integration step number k.

An additional issue is that $\Delta\tau_k$ may reach drastically small values in the region of the central peak of the electric impulse emitted in a dipole, *i.e.*, $1-\vec{n}(t)\cdot\vec{\beta}(t) \rightarrow 1/2\gamma^2$, whereas the total integrated time $\sum_{k=1}^{N}\Delta\tau_k$ may be several orders of magnitude larger. In terms of the physical phenomenon, the total duration of the electric field impulse as seen by the observer corresponds to the time delay $\sum_{k=1}^{N}\Delta\tau_k$ that separates photons emitted at the entrance of the magnet from photons emitted at the exit, but the significant part of it (in terms of energy density) which can be represented by the width $2\,\tau_c = 2\,\dfrac{2\rho}{3\gamma^3\,c}(1+\gamma^2\psi^2)^{3/2}$ of the radiation peak, is a very small fraction of $\sum_{k=1}^{N}\Delta\tau_k$.
The consequence is that, again in relation to computer precision, the differential element $\Delta\tau_k$ involved in the computation of eq. 3.2.11 cannot be derived from such relation as $\Delta\tau_k = \sum_{k=1}^{n}\Delta\tau_k - \sum_{k=1}^{n-1}\Delta\tau_k$ but instead must be stored as is, in the course of the ray-tracing process, for subsequent data treatment.

4 SPACE CHARGE

zgoubi provides models for the simulation of space charge effects. This capability has been first developped in the context of a PhD work regarding fixed field ring proton driver design studies and accelerator-driven subcritical reactor applications. This section of the guide is essentially drawn from that PhD dissertation [29], which can be referred to for additional details - the document provides benchmarking proofs in both KV and Gaussian density models (see below), including bunch transport in a drift, transport in a beam line, bunch tracking in an FFAG ring represented using either **zgoubi**'s *FFAG* keyword or magnetic field maps.

TO BE DOCUMENTED FURTHER

5 DESCRIPTION OF THE AVAILABLE PROCEDURES

5.1 Introduction

This chapter gives an inventory of the procedures available in **zgoubi**, their associated “keyword”, and a brief description of the way they function.

The chapter has been split into several sections. Sections 5.2 to 5.5 explain the underlying content - physics and numerical methods - behind the keywords, they are organized by topics :

- How to defined an object (a set of initial coordinates),
- Available options,
- Optical elements and procedures,
- Output procedures.

Section 5.6 addresses further a series of functionalities that may be accessed by means of special input data or flags.

5.2 Definition of an Object

The description of the object, *i.e.*, initial coordinates of the ensemble of particles, must be the first procedure in the **zgoubi** input data file, zgoubi.dat.

Several types of automatically generated objects are available, they are described in the following pages and include,

- non-random object, with various distributions : individual particles, grids, object for *MATRIX*, etc.
- Monte Carlo distribution, with various distributions as well : 6-D window, ellipsoids, etc.

A recurrent quantity appearing in these procedures is *IMAX*, the number of particles to be ray-traced. The maximum value allowed for *IMAX* can be changed at leisure in the include file `'MAXTRA.H'` where it is defined (that requires re-compiling **zgoubi**).

5.2.1 MCOBJET : Monte-Carlo generation of a 6-D object

MCOBJET generates a set of *IMAX* random 6-D initial conditions (the maximum value for *IMAX* is defined in the include file `'MAXTRA.H'`). It can be used in conjunction with the keyword *REBELOTE* which either allows generating an arbitrarily high number of initial conditions, or, in the hypothesis of a periodic structure, allows multi-turn tracking with initial conditions at pass number *IPASS* identified with conditions at end of pass number $IPASS - 1$.

The first datum in *MCOBJET* is the reference rigidity (a negative value is allowed)

$$BORO = \frac{p_0}{q} \text{ (kG.cm)}$$

Depending on the value of the next datum, *KOBJ*, the *IMAX* particles have their initial random conditions Y, T, Z, P, X and D (relative rigidity, $B\rho/BORO$) generated on 3 different types of supports, as described below.
Next come the data

$$KY, KT, KZ, KP, KX, KD$$

that specify the type of probability density for the 6 coordinates.
KY, KT, KZ, KP, KX can take the following values :

1. uniform density, $p(x) = 1/2\delta x$ if $-\delta x \leq x \leq \delta x$, $p(x) = 0$ elsewhere,
2. Gaussian density, $p(x) = \frac{1}{\delta x\sqrt{2\pi}} e^{-\frac{x^2}{2\delta x^2}}$,
3. parabolic density, $p(x) = \frac{3}{4\delta x}(1 - \frac{x^2}{\delta x^2})$ if $-\delta x \leq x \leq \delta x$, $p(x) = 0$ elsewhere.

KD can take the following values :

1. uniform density, $p(D) = 1/2\delta D$ if $-\delta D \leq D \leq \delta D$, $p(D) = 0$ elsewhere,
2. exponential density, $p(D) = N_0 \exp(C_0 + C_1 l + C_2 l^2 + C_3 l^3)$ with $0 \leq l \leq 1$ and $-\delta D \leq D \leq \delta D$,
3. $p(D)$ is determined by a kinematic relation, namely, with $T =$ horizontal angle, $D = \delta D * T$.

Next come the central values for the random sorting,

$$Y_0, \quad T_0, \quad Z_0, \quad P_0, \quad X_0, \quad D_0$$

namely, the probability density laws $p(x)$ ($x = Y, T, Z, P$ or X) and $p(D)$ described above apply to the variables $x - x_0$ ($\equiv Y - Y_0$, $T - T_0$, ...) and $D - D_0$ respectively. Negative value for D_0 is allowed (see section 5.6.12, page 184).

KOBJ = 1 : Random generation of *IMAX* particles in a hyper-window with widths (namely the half-extent for uniform or parabolic distributions ($KY, KT, ... = 1$ or 3), and the r.m.s. width for Gaussian distributions ($KY, KT, ... = 2$))

$$\delta Y, \quad \delta T, \quad \delta Z, \quad \delta P, \quad \delta X, \quad \delta D$$

Then follow the cut-off values, in units of the r.m.s. widths δY, δT, ... (used only for Gaussian distributions, $KY, KT, ... = 2$)

$$N_{\delta Y}, \quad N_{\delta T}, \quad N_{\delta Z}, \quad N_{\delta P}, \quad N_{\delta X}, \quad N_{\delta D}$$

The last data are the parameters

$$N_0, \quad C_0, \quad C_1, \quad C_2, \quad C_3$$

needed for generation of the D coordinate upon option $KD = 2$ (unused if $KD = 1,\ 3$) and a set of three integer seeds for initialization of random sequences,

$$IR1, \quad IR2, \quad IR3 \quad (\text{all} \simeq 10^6)$$

All particles generated by *MCOBJET* are tagged with a (non-S) character, for further statistic purposes (*e.g.*, with *HISTO, MCDESINT*).

KOBJ = 2 : Random generation of *IMAX* $= IY * IT * IZ * IP * IX * ID$ particles on a hyper-grid. The input data are the number of bars in each coordinate

$$IY, \quad IT, \quad IZ, \quad IP, \quad IX, \quad ID$$

the spacing of the bars

$$PY, \quad PT, \quad PZ, \quad PP, \quad PX, \quad PD$$

the width of each bar

$$\delta Y, \quad \delta T, \quad \delta Z, \quad \delta P, \quad \delta X, \quad \delta D$$

the cut-offs, used with Gaussian densities (in units of the r.m.s. widths)

$$N_{\delta Y}, \quad N_{\delta T}, \quad N_{\delta Z}, \quad N_{\delta P}, \quad N_{\delta X}, \quad N_{\delta D}$$

This is illustrated in Fig. 7.

The last two sets of data in this option are the parameters

$$N_0, \quad C_0, \quad C_1, \quad C_2, \quad C_3$$

needed for generation of the D coordinate upon option KD= 2 (unused if KD= 1, 3) and a set of three integer seeds for initialization of random sequences, $IR1$, $IR2$, and $IR3$ (all $\simeq 10^6$).
All particles generated by *MCOBJET* are tagged with a (non-S) character, for further statistic purposes (see *HISTO and MCDESINT*).

KOBJ = 3 : Distribution of *IMAX* particles in a 6-D ellipsoid (with possible momentum correlation via dispersion) defined by the three sets of data (one set per 2-D phase-space)

$$\begin{array}{llllll}
\alpha_Y, & \beta_Y, & \dfrac{\varepsilon_Y}{\pi}, & N_{\varepsilon_Y} & [, \quad N'_{\varepsilon_Y}, \text{ if } N_{\varepsilon_Y} < 0] & [, \quad D_Y, \ D_T] \\
\alpha_Z, & \beta_Z, & \dfrac{\varepsilon_Z}{\pi}, & N_{\varepsilon_Z} & [, \quad N'_{\varepsilon_Z}, \text{ if } N_{\varepsilon_Z} < 0] & [, \quad D_Z, \ D_P] \\
\alpha_X, & \beta_X, & \dfrac{\varepsilon_X}{\pi}, & N_{\varepsilon_X} & [, \quad N'_{\varepsilon_X}, \text{ if } N_{\varepsilon_X} < 0], &
\end{array}$$

where $\alpha_{X,Y,Z}$, $\beta_{X,Y,Z}$, $D_{Y,T}$, $D_{Z,P}$ are the ellipse parameters and the dispersions and derivatives, $\varepsilon_{X,Y,Z}/\pi$ are the *rms* emittances. This defines for each plane an elliptical frontier :

$$\frac{1+\alpha_Y^2}{\beta_Y} Y^2 + 2\alpha_Y YT + \beta_Y T^2 = \varepsilon_Y/\pi,$$

$$\frac{1+\alpha_Z^2}{\beta_Z} Z^2 + 2\alpha_Z ZP + \beta_Z P^2 = \varepsilon_Z/\pi,$$

$$\frac{1+\alpha_l^2}{\beta_l} l^2 + 2\alpha_l l\delta + \beta_l \delta^2 = \varepsilon_l/\pi,$$

with l the bunch length variable (unit is cm) and δ a short notation for $\delta p/p$, momentum spread.

Sorting cut-offs N_{ε_Y}, N_{ε_Z} and N_{ε_X} are accounted for if distributions are Gaussian ($KY, KT, ... = 2$) ; in this case *rms* values are

$$\sigma_Y = \sqrt{\beta_Y \times \varepsilon_Y/\pi} \text{ and } \sigma_T = \sqrt{\gamma_Y \times \varepsilon_Y/\pi},$$
$$\sigma_Z = \sqrt{\beta_Z \times \varepsilon_Z/\pi} \text{ and } \sigma_P = \sqrt{\gamma_Z \times \varepsilon_Z/\pi},$$
$$\sigma_l = \sqrt{\beta_l \times \varepsilon_l/\pi} \text{ (bunch length, unit cm), and } \sigma_\delta = \sqrt{\gamma_l \times \varepsilon_l/\pi} \text{ momentum spread.}$$

For uniform sorting, the distribution extends over the intervals, for respectively the radial, axial and longitudinal distribution,

$$\pm\sqrt{\varepsilon_Y/\pi \times \beta_Y} \text{ and } \pm\sqrt{\varepsilon_Y/\pi \times \gamma_Y},$$
$$\pm\sqrt{\varepsilon_Z/\pi \times \beta_Z} \text{ and } \pm\sqrt{\varepsilon_Z/\pi \times \gamma_Z},$$
$$\pm\sqrt{\varepsilon_l/\pi \times \beta_l} \text{ and } \pm\sqrt{\varepsilon_l/\pi \times \gamma_l}.$$

• N_ε may be negative. The meaning of this is for the sorting to fill up an elliptical ring that extends from $|N_\varepsilon|$ to N'_ε (rather than the inner region determined by the N_ε cut-off as discussed above, when $N_\varepsilon > 0$).

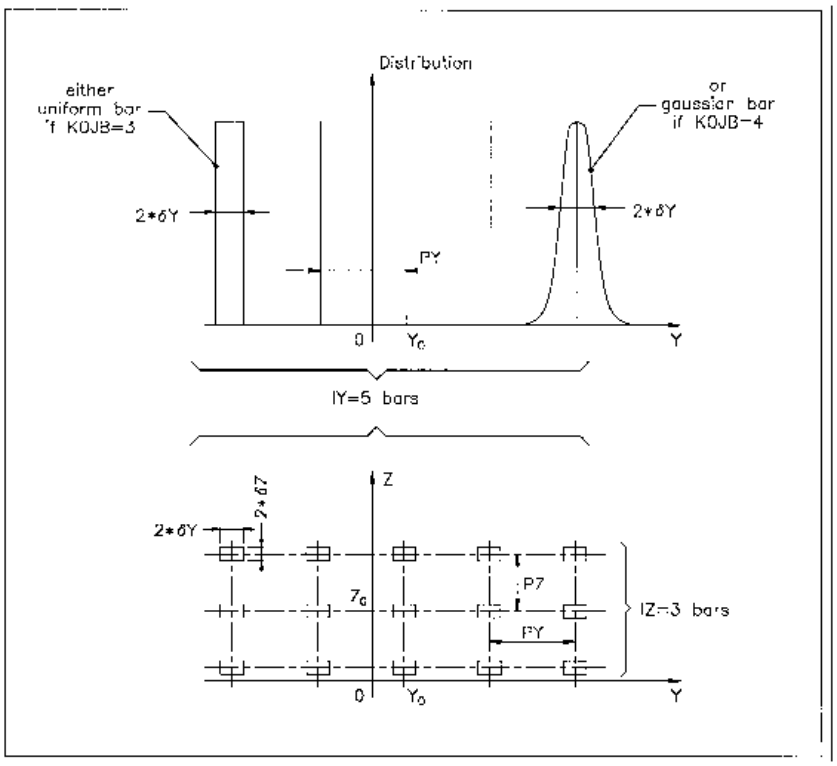


Figure 7: A scheme of input parameters to *MCOBJET* when *KOBJ*= 2.
Top : Possible distributions of the Y coordinate.
Bottom : A 2-D grid in (Y, Z) space.

5.2.2 OBJET : Generation of an object

OBJET is dedicated to the construction of sets of initial coordinates, in several ways.

The first datum is the reference rigidity (a negative value is allowed)

$$BORO = \frac{p_0}{q}$$

At the object, the beam is defined by a set of *IMAX* particles (the maximum value for *IMAX* is defined in the include file `'MAXTRA.H'`) with the initial conditions (Y, T, Z, P, X, D) with $D = B\rho/BORO$ the relative rigidity.
Depending on the value of the next datum *KOBJ*, these initial conditions may be generated in different ways, as follows.

KOBJ = 1 : Defines a grid in the Y, T, Z, P, X, D space. One defines the number of points desired

$$IY, \quad IT, \quad IZ, \quad IP, \quad IX, \quad ID$$

with $IY \leq n_Y \ldots ID \leq n_D$ such that $n_Y \times n_T \times ... \times n_D \leq max(IMAX)$. One defines the sampling range in each coordinate

$$dY, \quad dT, \quad dZ, \quad dP, \quad dX, \quad dD$$

zgoubi then generates $IMAX = IY * IT * IZ * IP * IX * ID$ particles with initial coordinates

$$\begin{array}{cccc}
0, & \pm dY, & \pm 2 * dY, & \ldots, \quad \pm IY/2 * dY, \\
0, & \pm dT, & \pm 2 * dT, & \ldots, \quad \pm IT/2 * dT, \\
0, & \pm dZ, & \pm 2 * dZ, & \ldots, \quad \pm IZ/2 * dZ, \\
0, & \pm dP, & \pm 2 * dP, & \ldots, \quad \pm IP/2 * dP, \\
0, & \pm dX, & \pm 2 * dX, & \ldots, \quad \pm IX/2 * dX, \\
0, & \pm dD, & \pm 2 * dD, & \ldots, \quad \pm ID/2 * dD,
\end{array}$$

In this option relative rigidities will be classified automatically in view of possible further use of *IMAGES*$[Z]$ for momentum analysis and image formation.
The particles are tagged with an index *IREP* possibly indicating a symmetry with respect to the (X,Y) plane, as explained in option *KOBJ*= 3. If two trajectories have mid-plane symmetry, only one will be ray-traced, while the other will be deduced using the mid-plane symmetries. This is done for the purpose of saving computing time. It may be incompatible with the use of some procedures (*e.g. MCDESINT*, which involves random processes).
The last datum is a reference in each coordinate, *YR*, *TR*, *ZR*, *PR*, *XR*, *DR*. For instance the reference rigidity is $DR*$ *BORO*, resulting in the rigidity of a particle defined by $I * dD$ to be $(DR + I * dD) * BORO$.

KOBJ = 1.1 : Same as *KOBJ*= 1 except for the Z symmetry. The initial Z and P conditions are the following

$$\begin{array}{cccc}
0, & dZ, & 2 * dZ, & \ldots, \quad (IZ - 1) * dZ, \\
0, & dP, & 2 * dP, & \ldots, \quad (IP - 1) * dP,
\end{array}$$

This object results in shorter outputs/CPU-time when studying problems with Z symmetry.

KOBJ = 2 : Next data : *IMAX*, *IDMAX*. Initial coordinates are entered explicitly for each trajectory. *IMAX* is the total number of particles. These may be classified in groups of equal number for each value of momentum, in order to fulfill the requirements of image calculations by *IMAGES*[*Z*]. *IDMAX* is the number of groups of momenta. The following initial conditions defining a particle are specified for each one of the *IMAX* particles

$$Y, \quad T, \quad Z, \quad P, \quad X, \quad D, \quad 'A'$$

where $D * BORO$ is the rigidity (negative value allowed) and $'A'$ is a (arbitrary) tagging character.

The last record *IEX* (I=1, *IMAX*) contains *IMAX* times either the character “1” to indicates that the particle has to be tracked, or “-9” to indicates that the particle should not be tracked.

This option *KOBJ*= 2 may be be useful for the definition of objects including kinematic effects.

KOBJ = 2.1 : Same as *KOBJ*= 2 except for the units, meter and radian in that case.

Note that in this option **KOBJ = 2[.]]**, only the first 7 trajectories can possibly have their coodinates varied when using the *FIT[2]* procedure.

KOBJ = 3 : This option allows the reading of initial conditions from an external input file *FNAME.*
The next three data lines are :

IT1, IT2, ITStep
IP1, IP2, IPStep
YF, TF, ZF, PF, SF, DPF, TiF, TAG
YR, TR, ZR, PR, SR, DPR, TiR
InitC

followed by the storage file name *FNAME.*

IT1, *IT2*, ITStep cause the code to read coordinates of particles number *IT1* through *IT2* by step *ITStep.*

IP1, IP2, IPStep cause the code to read coordinates belonging in the passes range *IP1* through *IP2*, step *IPStep.*

YF, TF, ZF, PF, SF, DPF, TiF are scaling factors whereas *YR, TR, ZR, PR, SR, DPR, TiR* are references added to the values of respectively Y, T, Z, P, S, DP as read in file *FNAME*, so that any coordinate $C = Y, T, Z...$ is changed into CF*C + CR. In addition a flag character TAG allows retaining only particles with identical tagging letter LET, unless TAG=’*’ in which case it has no selection effect - for instance TAG=’S’ can be used to retain only secondary particles following in-flight decay simulations.

If InitC$= 1$ ray-tracing starts from the current coordinates$F(J, I)$,
if InitC$= 0$ ray-tracing starts from the initial coordinates $FO(J, I)$, as read from file *FNAME.*

The file *FNAME* must be formatted in the appropriate manner. The following *FORTRAN* sequence is an instance, details and possible updates are to be found in the source file `'obj3.f'` :

```
        OPEN (UNIT = NL, FILE = FNAME, STATUS = ‘OLD’)
 DO I = 1, IMAX
    READ (NL,100) LET (I), IEX(I), (FO(J,I),J=1,6), (F(J,I),J=1,6), I, IREP(I),
     >      LET(I),IEX(I),-1.D0+FO(1,I),(FO(J,I),J=2,MXJ),
     >      -1.D0+F(1,I),F(2,I),F(3,I),
     >      (F(J,I),J=4,MXJ),ENEKI,
     >      ID,I,IREP(I), SORT(I),D,D,D,D,RET(I),DPR(I),
     >      D, D, D, BORO, IPASS, KLEY,LBL1,LBL2,NOEL
 100        FORMAT(1X,
C1   LET(IT),KEX,   1.D0-FO(1,IT),(FO(J,IT),J=2,MXJ),
     1           A1,1X,I2,1P,7E16.8,
C2   1.D0-F(1,IT),(FO(J,IT),J=2,MXJ),
     2         /,3E24.16,
C3   Z,P*1.D3,SAR,     TAR,     DS,
     3         /,4E24.16,E16.8,
C4   KART,  IT,IREP(IT),SORT(IT),X, BX,BY,BZ, RET(IT), DPR(IT),
     4         /,I1,2I6,7E16.8,
C5        EX,EY,EZ, BORO,  IPASS,  KLEY,  (LABEL(NOEL,I),I=1,2),NOEL
     5  /,4E16.8,          I6,1X,  A8,1X,  2A10,               I5)
  ENDDO
```

where the meaning of the parameters (apart from D=dummy real, ID=dummy integer) is the following

LET(I) : one-character string (for tagging)
IEX(I) : flag, see *KOBJ*= 2 and page 184
FO(1-6,I) : coordinates D, Y, T, Z, P and path length of particle number I, at the origin. $D * BORO$ = rigidity
F(1-6,I) : id, at the current position.

IREP is an index which indicates a symmetry with respect to the median plane. For instance, if $Z(I+1) = -Z(I)$, then normally $IREP(I+1) = IREP(I)$. Consequently the coordinates of particle $I+1$ will not be obtained from ray-tracing but instead deduced from those of particle I by simple symmetry. This saves on computing time.

KOBJ$= 3$ can be used for reading files filled by *FAISCNL, FAISTORE.*
If more than *IMAX* particles are to be read from a file, use *REBELOTE.*

Note : In this option, one has to make sure that input data do not conflict with possible use of the keyword *PARTICUL* that assigns mass and charge.

KOBJ = 3.1 : Same as **KOBJ = 3**, except for the formatting of trajectory coordinate data in *FNAME,* namely, according to the following *FORTRAN* sequence

```
      OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD')
 1    CONTINUE
      READ (NL,*,END=10,ERR=99) Y, T, Z, P, S, D
      GOTO 1
10    CALL ENDFIL
99    CALL ERREAD
```

KOBJ = 3.2 : As for **KOBJ=3.1**, except for the different format

```
READ(NL,*) X,Y,Z,PX,PY,PZ
```

where PX, PY, and PZ, are the momenta in MeV/c. Note that DPR will be ignored in this case.

KOBJ = 3.3 : As for **KOBJ=3.1**, except for the different format :

```
READ(NL,*) DP,Y,T,Z,P,S,TIME,MASS,CHARGE
```

where MASS is the mass in MeV/c and CHARGE is the charge in units of the elementary charge.

Note : For details and possible updates in the formatted read of concern in the *FORTRAN*, regarding options 3.1-3.3, see the source file `'obj3.f'`.

RECOVERING FROM A CRASH

OBJET with option *KOBJ*=3[.*] can be used to recover from a job crash, when multi-turn tracking - due to computer crash or else. This is discussed in the *SCALING* section, page 80, an example of job resuming is given page 255.

KOBJ = 5 : Mostly dedicated to the calculation of first order transfer matrix and various other optical parameters, using for instance *MATRIX* or *TWISS.* The user defines the coordinate sampling

$$dY, \quad dT, \quad dZ, \quad dP, \quad dD$$

These values should be small enough, so that the paraxial ray approximation be valid. The code generates 11 particles, with initial coordinates

$$0, YR \pm dY, TR \pm dT, ZR \pm dZ, PR \pm dP, DR \pm dD$$

given the reference trajectory

$$YR, \quad TR, \quad ZR, \quad PR, \quad DR$$

(with $DR * BORO$ the reference rigidity - negative value allowed).

KOBJ = 5.1 : Same as *KOBJ = 5*, except for an additional data line giving initial beam ellipse parameters and dispersions, $\alpha_Y, \beta_Y, \alpha_Z, \beta_Z, \alpha_X, \beta_X, D_Y, D'_Y, D_Z, D'_Z$, for further transport of these using *MATRIX* or *OPTICS*, or for possible use by the *FIT[2]* procedure.

KOBJ = 5.N ($N \geq 2$**) :** Like *KOBJ = 5* except for its allowing $N \geq 2$ reference trajectories, rather than just one as in *KOBJ = 5* case (thus *N-1* additional input data lines are needed in this case). These reference trajectories can be for instance chromatic orbits. Zgoubi will generate *N* sets of 11 particles with initial coordinates in each set taken *wrt.* one of the *N* references.

A subsequent use of *MATRIX* would then cause the computation of *N* transport matrices.

Note that in the option *KOBJ= 5.N,* $N \geq 2$, only the first 7 reference trajectories can possibly have their coodinates varied when using the *FIT[2]* procedure.

KOBJ = 6 : Mostly dedicated to the calculation of first, second and other higher order transfer coefficients and various other optical parameters, using for instance *MATRIX*. The input data are the coordinate sampling (normally taken paraxial)

$$dY, \quad dT, \quad dZ, \quad dP, \quad dD$$

to allow the building up of an object containing 61 particles (note : their coordinates can be checked by printing out into zgoubi.res using *FAISCEAU*), whereas a last data line gives the reference

$$YR, \quad TR, \quad ZR, \quad PR, \quad DR$$

(with $DR * BORO$ the reference rigidity - negative value allowed), which adds to the previous coordinate values.

KOBJ = 7 : Object with kinematics
The data and functioning are the same as for *KOBJ*= 1, except for the following

- ID is not used,
- PD is the kinematic coefficient, such that for particle number I, the initial relative rigidity D_I is calculated from the initial angle T_I following

  $$D_I = DR + PD * T_I$$

  while T_I is in the range

  $$0, \quad \pm PT, \quad \pm 2 * PT, \quad \ldots, \quad \pm IT/2 * PT$$

  as when using *KOBJ*= 1.

KOBJ = 8 : Generation of phase-space coordinates on ellipses.
The ellipses are defined by the three sets of data (one set per ellipse)

$$\begin{array}{lll} \alpha_Y, & \beta_Y, & \varepsilon_Y/\pi \\ \alpha_Z, & \beta_Z, & \varepsilon_Z/\pi \\ \alpha_X, & \beta_X, & \varepsilon_X/\pi \end{array}$$

where α, β are the ellipse parameters and $\varepsilon/$ is the ellipse surface, corresponding to an ellipse with equation

$$\frac{1+\alpha_Y^2}{\beta_Y} Y^2 + 2\alpha_Y YT + \beta_Y T^2 = \varepsilon_Y/\pi$$

(*idem* for the (Z, P) or (X, D) planes).

The ellipses are centered respectively on (Y_0, T_0), (Z_0, P_0), (X_0, D_0).

The number of samples per plane is respectively IX, IY, IZ. If that value is zero, the central value above is assigned.

5.2.3 OBJETA : Object from Monte-Carlo simulation of decay reaction [30]

This generator simulates the reactions

$$M_1 + M_2 \longrightarrow M_3 + M_4$$

and then

$$M_4 \longrightarrow M_5 + M_6$$

where M_1 is the mass of the incoming body ; M_2 is the mass of the target ; M_3 is an outgoing body ; M_4 is the rest mass of the decaying body ; M_5 and M_6 are decay products. Example :

$$p + d \longrightarrow {}^3\text{He} + \eta$$
$$\eta \longrightarrow \mu^+ + \mu^-$$

The first input data are the reference rigidity

$$BORO = p_0/q$$

an index *IBODY* which specifies the particle to be ray-traced, namely M3 (*IBODY* = 1), M5 (*IBODY* = 2) or M6 (*IBODY* = 3). In this last case, initial conditions for M6 must be generated by a first run of *OBJETA* with *IBODY* = 2 ; they are then stored in a buffer array, and restored as initial conditions at the next occurrence of *OBJETA* with *IBODY* = 3. Note that **zgoubi** by default assumes positively charged particles.

Another index, *KOBJ*, specifies the type of distribution for the initial transverse coordinates Y, Z ; namely either uniform (*KOBJ*= 1) or Gaussian (*KOBJ*= 2). The other three coordinates T, P and D are deduced from the kinematic of the reactions.

The next data are the number of particles to be generated, *IMAX*, the masses involved in the two previous reactions.

$$M_1, \quad M_2, \quad M_3, \quad M_4, \quad M_5, \quad M_6$$

and the kinetic energy T_1 of the incoming body (M_1).
Then one gives the central value of the distribution for each coordinate

$$Y_0, \quad T_0, \quad Z_0, \quad P_0, \quad D_0$$

and the width of the distribution around the central value

$$\delta Y, \quad \delta T, \quad \delta Z, \quad \delta P, \quad \delta D$$

so that only those particles in the range

$$Y_0 - \delta Y \leq Y \leq Y_0 + \delta Y \qquad \ldots \qquad D_0 - \delta D \leq D \leq D_0 + \delta D$$

will be retained. The longitudinal initial coordinate is uniformly sorted in the range

$$-XL \leq X_0 \leq XL$$

The random sequences involved may be initialized with different values of the two integer seeds IR_1 and IR_2 ($\simeq 10^6$).

Possible use of *PARTICUL* will have no effect : it will not change the mass and charge assumptions as set by *OBJETA*.

5.3 Declaring Options

A series of options are available which allow the control of various of the procedures and functionalities of the code.

Some of these options are normally declared right after the object definition, for instance

- *SPNTRK* : switch-on spin tracking,
- *PARTICUL* to declare particle mass and charge, if for instance tracking in electric fields, or tracking spin, or in presence of synchrotron radiation energy loss simulations,

some may appear further down in the structure (in zgoubi.dat), for instance

- *MCDESINT* : switch-on in-flight decay, could be after a target,
- *REBELOTE* : for multi-turn tracking, including an extraction line section for instance,

others may normally be declared at the end of zgoubi.dat data pile, for instance

- *END* : end of a problem,
- *FIT* : fitting procedure - can also appear before *REBELOTE*

GETFITVAL may appear *before* the object definition (keyword *[MC]OBJET*, normally the first in **zgoubi** input data list). This is the case if variables prior saved following a *'FIT[2]'* procedure and then read back using *GETFITVAL*, happen to belong in the *[MC]OBJET* data list.

SYSTEM as well, may appear anywhere in the data list.

5.3.1 BINARY : *BINARY/FORMATTED* data converter

This procedure translates field map data files from “BINARY” to “FORMATTED” – in the *FORTRAN* sense, or the other way.

The keyword is followed by, next data line,

$$NF[.J], NCOL, NHDR$$

the number of files to be translated [`READ` format option, a single digit integer, optional], number of data columns in the file, number of header lines in the file.

Fortran READ format, case of formatted input file :

If J is not given, the $NCOL$ arrangement should be consistent with the following *FORTRAN* READ statement :

```
READ (unit=ln,*) (X7(I),I=1,NCOL)
```

If $J = 1$, NCOL should be consistent with the following *FORTRAN* READ statement :

```
READ (unit=ln,fmt='(1x,ncol*E11.2)') (X7(I),I=1,NCOL)
```

Fortran READ format, case of formatted input file :

Data are expected to have been recorded under the form of Ncol column rows.

Then follow, line by line, the NF names of the files to be translated.

Iff a file name begins with the prefix “B_” or “b_”, it is assumed “binary”, and hence converted to “formatted”, and given the same name after suppression of the prefix “B_” or “b_”. Conversely, *iff* the file name does not begin with “B_” or “b_”, the file is presumed “formatted” and hence translated to “binary”, and is given the same name after addition of the prefix “b_”.

In its present state, the procedure *BINARY* only supports a limited number of read/write formats. Details concerning I/O formatting can be found in the *FORTRAN* file `'binary.f'`.

5.3.2 END or FIN : End of input data list

The end of a problem, or of a set of several problems stacked in the data file, should be stated by means of the keywords *FIN* or *END.*

Any information following these keywords will be ignored.

In some cases, these keywords may cause some information to be printed in zgoubi.res, for instance when the keyword *PICKUPS* is used.

5.3.3 ERRORS : Injecting errors in optical elements

UNDER DEVELOPMENT

The keyword *ERRORS* allows injecting diverse types of errors in optical elements, including field and positionning defects. Uniform and Gaussian distribution laws are available.

Optical elements concerned can be discriminated using one or both of their labels.

ERRORS is under development, functionalities at the moment are the following :

MULTIPOL can be injected multipole field defects on any one of its dipole to 20-pole component. The defect can be relative (to the current field value) or absolute (adds to the current field value).

TOSCA can be injected a field defect that applies to the field map as a whole. The defect can be relative (to the current field value) or absolute (adds to the current field value).

Some rules :

- if *REBELOTE* is used in multi-turn mode (*i.e.*, its K argument value is 99), then the same defect series is applied at each pass, by default. However one may want to simulate one turn by repeating a cell subset, using *REBELOTE*, in which case defects of random type in the subset should be changed at each pass. This requires modification in errors.f program.

5.3.4 FIT, FIT2 : Fitting procedure

The keywords *FIT, FIT2* allow the automatic adjustment of up to 20 variables, for fitting up to 20 constraints.

FIT was implemented in 1985, drawn from the matrix transport code BETA [31]. *FIT2* is a simplex method (Nelder-Mead method), it has been implemented in 2007 [32]. One or the other may converge faster, or may have some advantages/disadvantages, depending on the problem.

Any physical parameter of any element in the zgoubi.dat data list may be varied. Examples of available constraints are, amongst others :
- trajectory coordinates in the $F(J, I)$ array, I = particle number, J = coordinate number = 1 to 7 for respectively D, Y, T, Z,P, S =path length, time ;
- spin coordinates ;
- any of the 6×6 coefficients of the first order transfer matrix $[R_{ij}]$ as defined in the keyword *MATRIX* ;
- any of the $6 \times 6 \times 6$ coefficients of the second order array $[T_{ijk}]$ as defined in *MATRIX* ;
- any of the 4×4 coefficients of the beam σ-matrix
- transmission efficiency of an optical channel.
- tunes $\nu_{Y,Z}$ and periodic betatron functions $\beta_{Y,Z}, \alpha_{Y,Z}, \gamma_{Y,Z}$, as computed in the coupled hypothesis [33].

A full list of the constraints available is given in the table page 233.

FIT, FIT2 are compatible with the use of (*i.e.*, can be encompassed in) *REBELOTE* for successive fitting trials using various sets of parameters (option $K = 22$ in *REBELOTE*).

VARIABLES

The first input data in *FIT[2]* is the number of variables *NV*. A variable is defined by a line of data comprised of

$IR =$ number of the varied element in the structure
$IP =$ number of the physical parameter to be varied in this element
$XC =$ coupling parameter. Normally $XC = 0$. If $XC \neq 0$, coupling will occur (see below)

followed by, either

$DV =$ allowed relative range of variation of the physical parameter IP

or

$[Vmin, Vmax] =$ allowed interval of variation of the physical parameter IP

Numbering of the Elements (*IR*) :

The elements (*i.e.*, keywords *DIPOLE, QUADRUPO,* etc.) as read by **zgoubi** in the zgoubi.dat sequence are assigned a number. which follows their sequence in the data file. It is that very number, *IR*, that the *FIT[2]* procedure uses. A simple way to get IR once the zgoubi.dat file has been built, is to do a preliminary run, since the first thing **zgoubi** does is copy the sequence from zgoubi.dat into the result file zgoubi.res, with all elements numbered.

Numbering of the Physical Parameters (*IP*) :

All the data that follow a keyword are numbered - except for *SCALING*, see below.

With most of the keywords, the numbering follows the principle hereafter :

Input data	**Numbering for FIT**
'KEYWORD'	
first line	1, 2, 3, ..., 9
second line	10, 11, 12, 13, ..., 19
this is a comment	a line of comments is skipped
next line	20, 21, 22, ..., 29
and so on...	

The examples of *QUADRUPO* (quadrupole) and *TOSCA* (Cartesian or cylindrical mesh field map) are as follows.

Input data	**Numbering for FIT**
'QUADRUPO'	
IL	1
XL, R_0, B	10, 11, 12
X_E, λ_E	20, 21
NCE, C_0, C_1, C_2, C_3, C_4, C_5	30, 31, 32, 33, 34, 35, 36
X_S, λ_S	40, 41
NCS, C_0, C_1, C_2, C_3, C_4, C_5	50, 51, 52, 53, 54, 55, 56
XPAS	60
KPOS, XCE, YCE, ALE	70, 71, 72, 73
'TOSCA'	
IC, IL	1, 2
BNORM, X- [, Y-, Z-]NORM	10, 11 [, 12, 13]
TIT	This is text
IX, IY, IZ, *MOD*	20, 21, 22, 23
FNAME	This is text
ID, A, B, C [A', B', C', etc. if $ID \geq 2$]	30, 31, 32, 33 [34, 35, 36 [, 37, 38, 39] if $ID \geq 2$]
IORDRE	40
XPAS	50
KPOS, XCE, YCE, ALE	60,61,62,63

A different numbering, fully sequential, has been adopted in the following elements :

AIMANT, *DIPOLE*, *EBMULT*, *ELMULT*, *MULTIPOL*.

It is illustrated here after in the case of *MULTIPOL* and *DIPOLE-M.*

Input data	**Numbering for FIT**
'MULTIPOL'	
0	1
365.760 10.0 7.5739 1.4939 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0	2, 3, 4, 5, ..., 13
10.0 4.0 0.80 0.0 0.0 0.0 0.0 0. 0. 0. 0.	14, 15, ..., 24
NC, C_0, C_1, C_2, C_3, C_4, C_5	25, 26, 27, 28, 29, 30, 31
10.0 4.0 0.80 0.0 0.0 0.0 0.0 0. 0. 0. 0.	32, 33, ..., 42
NC, C_0, C_1, C_2, C_3, C_4, C_5	43, 44, ..., 49
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.	50, 51, ..., 59
step size	60 *[and 61, 62 reserved]*
KPOS, XCE, YCE, ALE	63, 64, 65, 66

Input data	**Numbering for FIT**
'DIPOLE-M'	
NFACE, IC, IL	1, 2, 3
IAMAX, IRMAX	4, 5
B_0, N, B, G	6, 7, 8, 9
AT, ACENT, RM, RMIN, RMAX	10, 11, 12, 13, 14
λ , ξ	15,16
NC, C_0, C_1, C_2, C_3, C_4, C_5, shift	17, 18, 19, 20, 21, 22, 23, 24
ω, θ, R_1, U_1, U_2, R_2	25, 26, 27, 28, 29, 30
etc.	

Parameters in *SCALING* also have a sequential numbering, yet some positions are skipped, this is illustrated in the example hereafter which covers all possible working modes of *SCALING* (all details regarding the numbering can be found in the *FORTRAN* subroutine `rscal.f`) :

Input data	**Numbering in FIT**	**Quantities to be varied** (see SCALING for details)
'SCALING'		
1 9	1 2	Non relevant
AGSMM *AF *BF		Keywords concerned, their labels
-1 3 12 1. 13 1. 14 1.	3 4 5	dB1, dB2, dB3 parameters in AGSMM
7.2135	6	Field factor
1	7	Timing
AGSMM *AD *BD		
-1 3 12 1. 13 1. 14 1.	8 9 10	
7.2135	11	
1	12	
AGSMM *CF		
-1 3 12 1. 13 1. 14 1.	13 14 15	
7.2135	16	
1	17	
AGSQUAD QH_*		
3		
0.605 0.77 0.879	18 19 20	Field factor
1 2000 10000	21 22 23	Timing
AGSQUAD QV_*		
3		
0.587 0.83 0.83	24 25 26	
1 2000 10000	27 28 29	
MULTIPOL		General rule. Precedes labeled cases if any.
-1	30	(meaningless)
0.72135154291	31	
1	32	(meaningless)
MULTIPOL COH1		
1.10		No numbering if 1.10 type of option
./Csnk3D/bump_centered.scal		
1 2		
MULTIPOL COH2		
1.10		
./Csnk3D/bump_centered.scal		
1 4		
MULTIPOL KICKH KICKV		
2		
0.1 0.3	33 34	Field factor
1 10	35 36	Timing

Coupled Variables (XC)

Coupling a variable parameter to any other parameter in the structure is possible. This is done by giving XC a value of the form $int.ijk$ where the integer part "int" is the number of the coupled element in the structure (equivalent to IR, see above), and the decimal part "ijk" is the number of its parameter of concern (equivalent to IP, see above) (if the parameter number is in the range 1, 2, ..., 9 (resp. 10, 11, ... 19, and 100, ...), then ijk must take the form $00k$ (resp. $0jk$, ijk)). For example, $XC = 20.010$ is a request for coupling with the parameter number 10 of element number 20 of the structure, while $XC = 20.100$ is a request for coupling with the parameter number 100 of element 20.

An element of the structure which is coupled (by means of $XC \neq 0$) to a variable declared in the data list of the *FIT[2]* keyword, needs not appear as one of the *NV* variables in that data list (this would be redundant information).

XC can be either positive or negative. If $XC > 0$, then the coupled parameter will be given the same value as the variable parameter (for example, symmetric quadrupoles in a lens triplet will be given the same field). If $XC < 0$, then the coupled parameter will be given a variation opposite to that of the variable, so that the sum of the two parameters stays constant (for example, an optical element can be shifted while preserving the length of the structure, by coupling together its upstream and downstream drift spaces).

Variation Range

There are two ways to define the allowed range for a variable, as follows.

(i) DV : For a variable (parameter number IP under some keyword) with initial value v, the *FIT[2]* procedure is allowed to explore the range $v \times (1 \pm DV)$.

(i) $[v_{min}, v_{max}]$: This specifies the allowed interval of variation.

CONSTRAINTS

The next input data in *FIT[2]* is the number of constraints, *NC*. A list of the available constraints is given in the table page 233 ; adding or changing a constraint resorts to the *FORTRAN* file `ff.f`.

Each constraint is defined by the following list of data :

$IC[.IC2]$	=	type of constraint (see table p. 233).
I, J	=	constraint (*i.e.*, R_{ij}, determinant, tune ; T_{ijk} ; σ_{ij} ; trajectory $\#I$ and coordinate $\#J$)
IR	=	number of the keyword at the exit of which the constraint applies ; instead of numeral, can also be '#End', so the constraint applies at the last keyword preceeding 'FIT[2]'
V	=	desired value of the constraint
W	=	weight of the constraint (smaller W for higher weight)
NP		*NP* additional parameters, values PR_{1-NP}, defining the constraint

IC=0 : The coefficients σ_{11} (σ_{33}) = horizontal (vertical) beta values and σ_{22} (σ_{44}) = horizontal (vertical) derivatives ($\alpha = -\beta'/2$) are obtained by transport of their initial values at line start as introduced using for instance *OBJET, KOBJ=5.1.*

IC=0.1 : Beam parameters : $\sigma_{11} = \beta_Y, \sigma_{12} = \sigma_{21} = -\alpha_Y, \sigma_{22} = \gamma_Y$, $\sigma_{33} = \beta_Z, \sigma_{34} = \sigma_{43} = -\alpha_Z, \sigma_{44} = \gamma_Z$; periodic dispersion : $\sigma_{16} = D_Y$, $\sigma_{26} = D'_Y$, $\sigma_{36} = D_Z, \sigma_{46} = D'_Z$, all quantities derived by assuming periodic structure and identifying the first order transfer matrix with the form $Icos\mu + Jsin\mu$.

IC=1, 2 : The coefficients R_{ij} and T_{ijk} are calculated following the procedures described in *MATRIX*, option *IFOC* = 0. The fitting of the $[R_{ij}]$ matrix coefficients supposes the tracking of particles with paraxial coordinates, normally defined using *OBJET* option *KOBJ* = 5 or 6.

Type of constraint	Parameters defining the constraints						Recommended *[MC]OBJET* ; comments
	IC	I	J	Constraint	Additional parameter(s) NP	Additional parameter(s) Param. values, $pr_1 - pr_{NP}$	
Transported σ-matrix $(\sigma(s) = T\sigma(0)\tilde{T})$	0	1 - 6	1 - 6	σ_{IJ} $(\sigma_{11} = \beta_Y, \sigma_{21,\,12} = \alpha_Y$, etc.)			*OBJET/KOBJ=5.1*
Periodic σ-matrix $(\sigma = I\cos\mu + J\sin\mu)$ (N=1-9 for *MATRIX* block 1-9)	0.N $(N \leq 9)$	1 - 6 7 8 9 10	1 - 6 any any any any	σ_{IJ} $(\sigma_{11} = \cos\mu_Y + \alpha_Y \sin\mu_Y$, etc.) $\mu_Y/2\pi$ $\mu_Z/2\pi$ $\cos(\mu_Y)$ $\cos(\mu_Z)$			*OBJET/KOBJ=5.N*
First order transport coeffs.	1	$1-6$ 7 8	$1-6$ i j	Transport coeff. R_{IJ} $i \neq 8$: YY-determinant ; i=8 : YZ-det. $j \neq 7$: ZZ-determinant ; j=7 : ZY-det.			*OBJET/KOBJ=5*
Second order transport coeffs.	2	$1-6$	$11-66$	Transport coeff. $T_{I,j,k}$ $(j = [J/10], k = J - 10[J/10])$			*OBJET/KOBJ=6*
Trajectory	3	$1-IMAX$	$1-7$	$F(J,I)$			*[MC]OBJET*
coordinates		-1	$1-7$	$<F(J,i)>_{i=I1,I2}$	$\{0$ $\{2$	 $I_1, \quad I_2$	$1 \rightarrow IMAX$ $1 \leq I_1 \leq I_2 \leq IMAX$
(I = particle number; J=1-7 for D,Y,T,Z,P,S,time)		-2	$1-7$	$Sup(\lvert F(J,i)\rvert)_{i=1,IMAX}$			
		-3	$1-7$	$Dist\ \ F(J,I)_{i=I1,I2}$	3	$I_1, I_2, \Delta I$	$1 \leq I_1 \leq I_2 \leq IMAX$
		-4	$1-7$	$Dist\,[PU_i, i = 1, N]$	2	$NOEL_A$, $NOEL_B$	PU range
	3.1	$1-IMAX$	$1-7$	$\lvert F(J,I) - FO(J,I)\rvert$			
	3.2	$1-IMAX$	$1-7$	$\lvert F(J,I) + FO(J,I)\rvert$			
	3.4	$1-IMAX$	$1-7$	$\lvert F(J,I) - F(J,K)\rvert$	1	K	$K \leq IMAX$
	3.5	$1-IMAX$	$1-7$	$(F(J,I) - F(J,K))/F(J,K)$	1	K	$K \leq IMAX$
Ellipse parameters	4	$1-6$	$1-6$	σ_{IJ} $(\sigma_{11} = \beta_Y,$ $\sigma_{12} = \sigma_{21} = \alpha_Y$, etc.)			*OBJET/KOBJ=8 ;* *MCOBJET/KOBJ=3*
Number of particles	5	-1 $1-3$ $4-6$	any any any	$N_{survived}/IMAX$ $N_{in\ \epsilon_{Y,Z,X}}/N_{survived}$ $N_{in\ best\ \epsilon_{Y,Z,X,rms}}/N_{survived}$	 1 	 $\epsilon_{Y,Z,X}/\pi$ 	*OBJET* *MCOBJET* *MCOBJET*
Coordinates & fields, across optical elements (J=1, 2, 3 for $B_{X,\,Y,\,Z}$)	7.1 7.2 7.3 7.6 7.7 7.8 7.9	$1-IMAX$ $1-IMAX$ $1-IMAX$ $1-IMAX$ $1-IMAX$ $1-IMAX$ $1-IMAX$	$1-7$ $1-7$ $1-7$ $1-3$ $1-3$ $1-3$ $1-3$	min. ($pr_1 = 1$) or max. (2) of $F(J,I)$ max($F(J,I)$) - min$F(J,I)$) min$F(J,I)$) + max($F(J,I)$) min. ($pr_1 = 1$) or max. (2) value of B_J max(B_J) - min(B_J) min(B_J) + max(B_J) $\int B_J\, ds$	1 1 1 1 1 1 1	1-2 1-2 1-2 1-2 1-2 1-2 1-2	*[MC]OBJET*
Spin	10 10.1 10.2 10.3	$1-IMAX$ $1-IMAX$ $1-IDMAX$ $1-IDMAX$	$1-4$ $1-3$ any 1-3	$S_{X,Y,Z}(I)$, $\lvert\vec{S}(I)\rvert$ $\lvert S_{X,Y,Z}(I) - SO_{X,Y,Z}(I)\rvert$ spin rotation angle (rad) rotation axis, X-, Y- or Z-component			*SPNTRK+* *[MC]OBJET* *OBJET/KOBJ=2* *OBJET/KOBJ=2*

IMAX: total number of particles tracked.
F(J,I) : particle coordinate array ; J : coordinate (1-7 for respectively D, Y, T, Z, P, S, time, resp. units : none, cm, mrad, cm, mrad, cm, μs) ; I : particle number.
FO(J,I) : initial particle coordinates.
S(J,I) : spin coordinate array ; J : coordinate (1, 2, 3 for respectively X, Y, Z) ; I : particle number.
SO(J,I) : initial spin coordinates.

IC=3 : If $1 \leq I \leq IMAX$ then the value of coordinate type J ($J = 1, 6$ for respectively $D,\ Y,\ T,\ Z,\ P,\ S$) of particle number I ($1 \leq I \leq IMAX$) is constrained. However I can take special meaning, as follows.

$I = -1$: the constraint is the average value of coordinate of type J over particles $[I_1, I_2]$,

$I = -2$: the constraint is the maximum value of coordinate of type J,

$I = -3$: the constraint is the distance between particles $I_1 - I_2$ (step ΔI).

$I = -4$: take the average of the $F(J,i)$ coordinate (J=1-7 for D,Y,T,Z,P,S,time) over $(I2 - I1 + 1)$ particles in the range $i = I1 - I2$, get it at N different pickup locations $PU_1, PU_2, ..., PU_N$. The constraint is, minimizing the distance between these averages, $Dist\left[< F(J,i) >_{i=I1,I2}\right]_{PU_{1-N}}$ (Ex. : minimize the difference between PU records, for a series of PUs along a beamline). If the number of locations is 997 (*i.e.*, N+2=999) then all PUs upstream of IR are accounted for. This constraint requires declaring pick-ups using *PICKUPS* keyword, their locations are as numbered in the data list in zgoubi.res.

IC=3.1 : Absolute value of the difference between local and initial J-coordinate of particle I (convenient *e.g.* for closed orbit search).

IC=3.2 : Absolute value of the sum of the local and initial J-coordinate of particle number I.

IC=3.4 : Absolute value of the difference between local J-coordinates of particles respectively I and K.

IC=3.5 : Difference between local J-coordinates of particles respectively I and K, relative to J-coordinate of particle K.

IC=4 : The coefficients σ_{11} (σ_{33}) = horizontal (vertical) beta values and σ_{22} (σ_{44}) = horizontal (vertical) derivatives ($\alpha = -\beta'/2$) are derived from an ellipse match of the current particle population (as generated for instance using *MCOBJET, KOBJ=3*).
The fitting of the $[\sigma_{ij}]$ coefficients supposes the tracking of a relevant population of particles within an appropriate emittance.

IC=5 : The constraint value is the ratio of particles (over *IMAX*). Three cases possible :

$I = -1$, ratio of particles still on the run.

$I = 1,\ 2,\ 3$, *maximization* of the number of particles encompassed within a given I-type (for respectively $Y,\ Z,\ D$) phase-space emittance value. Then, *NP*=1, followed by the emittance value. The center and shape of the ellipse are determined by a matching to the position and shape of the particle distribution.

$I = 4,\ 5,\ 6$, same as previous case, except for the ellipse, taken to be the *rms* matched ellipse to the distribution. Thus *NP*=0.

IC=7 series : The quantity to be constrained is taken *inside* the optical element with number *IR* (see page 65).

IC=7.1 : Minimum or maximum value as reached by coordinate J of particle number I.

IC=7.2 : Maximum minus minimum of the value of coordinate J of particle number I.

IC=7.3 : Maximum plus minimum of the value of coordinate J of particle number I.

IC=7.6 : Minimum (*PR* $= 1$) or maximum (*PR* $= 2$) value as reached by the X, Y or Z component of the field along the trajectory of particle number I.

IC=7.7 : Maximum minus minimum of the values taken by the X, Y or Z component of the field along the trajectory of particle number I.

IC=7.8 : Maximum plus minimum of the values taken by the X, Y or Z component of the field along the trajectory of particle number I.

IC=7.9 : Integral of the X, Y or Z component of the field as experienced by particle number I, across the optical element.

IC=10 : If $1 \leq I \leq$ *IMAX* then the constrint is the value of coordinate type J ($J = 1, 3$ for respectively S_X, S_Y, S_Z) of particle number I.

IC=10.1 : Difference between final and initial J-spin coordinate of particle I (convenient *e.g.* for $\vec{n}_0$ spin vector search).

IC=10.2 : Constraint on spin rotation angle for particles of momentum group #K. This requires *OBJET/KOBJ=2*, with groups of 3 particles, all particles in a group have the same momentum, and their spins in direction, respectively, X, Y, Z.

IC=10.3 : Constraint on spin rotation axis for particles of momentum group #K. This requires *OBJET/KOBJ=2*, with groups of 3 particles, all particles in a group have the same momentum, and their spins in direction, respectively, X, Y, Z.

OBJECT DEFINITION

Depending on the type of constraint (see table p. 233), constraint calculations are performed either from transport coefficient calculation and in such case require *OBJET* with either *KOBJ* = 5 or *KOBJ* = 6, or from particle distributions and in this case need object definition using for instance *OBJET* with *KOBJ* = 8, *MCOBJET* with *KOBJ* = 3.

THE FITTING METHODS

The *FIT* procedure was drawn from the matrix transport code BETA [31]. It is a direct sequential minimization of the quadratic sum of all errors (*i.e.*, differences between desired and actual values for the *NC* constraints), each normalized by its specified weight W (the smaller W, the stronger the constraint).
The step sizes for the variation of the physical parameters depend on their initial values, and cannot be accessed by the user. At each iteration, the optimum value of the step size, as well as the optimum direction of variation, is determined for each one of the *NV* variables. Then follows an iterative global variation of all *NV* variables, until the minimization fails which results in a next iteration on the optimization of the step sizes.

The *FIT2* procedure is based on the Nelder-Mead method, it has various specificities, details can be found in Ref. [32].

The optimization process may be stopped by means of a penalty value, or a maximum number of iterations on the step size or on the call to the function.

COMBINING FIT[2] AND REBELOTE

FIT[2] may be followed by the keyword *REBELOTE*. This allows for instance executing again the fitting procedure, following a change, by *REBELOTE*, in the value of some parameter(s) in zgoubi.dat data list. More on that can be found under *REBELOTE*, Sec. 5.3.12, page 78.

An example is given page 234.

Conversely, *REBELOTE* may be followed by the keyword *FIT[2].* This allows for instance optimization of constraints that require multi-turn tracking (*e.g.*, optimized tune-jump quadrupole settings for beam acceleration through depolarizing resonances).

FIT[2] OPTIONS

FIT and *FIT2* accept various options, as follows.

- 'nofinal' option : a request to avoid a final run that would normally follow the completion of the fit, allowing for the variable values as just determined by *FIT[2].* By default, *i.e.*, if 'nofinal' does not appear in the *FIT[2]* data list, then that final run does occur.

- 'save' [, FileName] option : a request for the variable values as resulting from *FIT[2]* to be saved, by default in zgoubi.FITVALS.out, or in *FileName* file if the latter is specified following the 'save' instruction.

- Penalty : value of the penalty to be reached, following what the *FIT[2]* procedure will be considered completed, and thus stopped.

- ITER : maximum allowed number of "calls to the function" (namely, (recursive) calls to zgoubi). Once that number is reached, *FIT[2]* is stopped, regardless of a possible "penalty" value.

An example using all five options, "nofinal", "save [FileName]", "penalty" and "ITER", is given in page 235.

5.3.5 GASCAT : Gas scattering

Modification of particle momentum and velocity vector, performed at each integration step, under the effect of scattering by residual gas.

Installation is to be completed.

5.3.6 GETFITVAL : Get values of *variables* as saved from former FIT[2] run

This keyword allows reading, from a file whose name needs be specified, parameter values to be assigned to optical elements in zgoubi.dat.

That file is expected to contain a copy-paste of the data under the *FIT[2]* procedure as displayed in zgoubi.res, normally under the form

```
STATUS OF VARIABLES  (Iteration #    95)
 LMNT  VAR PARAM  MINIMUM    INITIAL    FINAL      MAXIMUM     STEP    NAME       LBL1 LBL2
  145    1   4    -3.000E+03    762.   761.9484791 3.000E+03  1.254E-05 MULTIPOL   HKIC DHCB02
  182    2   4    -1.000E+03   -231.  -230.9846875 1.000E+03  4.182E-06 MULTIPOL   HKIC DHCB08
  146    3   4    -1.000E+03   -320.  -319.8554171 1.000E+03  4.182E-06 MULTIPOL   VKIC DVCB02
  183    4   4    -1.000E+03    528.   527.7249064 1.000E+03  4.182E-06 MULTIPOL   VKIC DVCB08
  615    5   4    -3.000E+03    308.   307.6860565 3.000E+03  1.254E-05 MULTIPOL   HKIC DHCF02
  651    6   4    -1.000E+03   -114.  -113.8490362 1.000E+03  4.182E-06 MULTIPOL   HKIC DHCF08
  616    7   4    -1.000E+03   -78.9  -78.88730937 1.000E+03  4.182E-06 MULTIPOL   VKIC DVCF02
  652    8   4    -1.000E+03    212.   211.8789183 1.000E+03  4.182E-06 MULTIPOL   VKIC DVCF08
# STATUS OF CONSTRAINTS
# TYPE  I   J  LMNT#     DESIRED       WEIGHT         REACHED         KI2     *  Parameter(s)
#   3   1   2   127   0.0000000E+00  1.0000E+00    1.0068088E-08   6.0335E-01 *  0 :
#   3   1   3   127   0.0000000E+00  1.0000E+00    7.0101405E-09   2.9250E-01 *  0 :
#   3   1   4   127   0.0000000E+00  1.0000E+00    2.9184383E-10   5.0696E-04 *  0 :
#   3   1   5   127   0.0000000E+00  1.0000E+00    3.1142381E-10   5.7727E-04 *  0 :
#   3   1   2   436   0.0000000E+00  1.0000E+00    3.8438378E-09   8.7944E-02 *  0 :
#   3   1   3   436   0.0000000E+00  1.0000E+00    1.5773011E-09   1.4808E-02 *  0 :
#   3   1   4   436   0.0000000E+00  1.0000E+00    2.2081272E-10   2.9022E-04 *  0 :
#   3   1   5   436   0.0000000E+00  1.0000E+00    5.7930552E-11   1.9975E-05 *  0 :
#   Function  called   1859 times
# Xi2 =    1.68006E-16   Busy...
```

A ’#’ at the beginning of a line means it is commented, thus it will not be taken into account. However a copy-paste from zgoubi.res (which is the case in the present example) would not not need any commenting.

Since some of the *FIT[2]* variables may belong in *[MC]OBJET*, *GETFITVAL* may appear upstream of *[MC]OBJET* in zgoubi.dat, to allow updating the latter.

Once completed, *FIT[2]* will be followed by a final run of zgoubi.dat with variable values updated as resulting from the fit. This can be inhibited by indicating ‘nofinal’ option in *FIT[2]* (see page 68). For that final run *GETFITVAL* will be inhibited so avoid overridding the updated variable values.

5.3.7 MCDESINT : Monte-Carlo simulation of in-flight decay[34]

When *MCDESINT* is met in a structure (normally, after *OBJET* or after *CIBLE*), in-flight decay simulation starts. It must be preceded by *PARTICUL* for the definition of mass M_1 and *COM* lifetime τ_1 of the parent particle.
The two-body decay simulated is

$$1 \longrightarrow 2 + 3$$

The decay is isotropic in the center of mass. 1 is the incoming particle, with mass M_1, momentum $p_1 = \gamma_1 M_1 \beta_1 c$ (relative momentum $D_1 = \dfrac{p_1}{q}\dfrac{1}{BORO}$ with *BORO* = reference rigidity, defined in *[MC]OBJET*). 2 and 3 are decay products with respective masses and momenta M_2, M_3 and $p_2 = \gamma_2 M_2 \beta_2 c$, $p_3 = \gamma_3 M_3 \beta_3 c$. The average distance s_1 at which a particle will decay is related to its center of mass lifetime τ_1 by

$$s_1 = c\tau_1\sqrt{\gamma_1^2 - 1}$$

The actual path length s up to the decay point is calculated, for each one of the particles defined by *[MC]OBJET* and prior t ray-tracing, from a random number $0 < R_1 \leq 1$ by using the exponential decay formula

$$s = -s_1 \ell n R_1$$

After decay of the parent particle 1, particle 2 will be ray-traced with assumed positive charge, while particle 3 is discarded. Its scattering angles in the center of mass θ^* and ϕ are generated from two other random numbers $0 < R_2 \leq 1$ and $0 < R_3 \leq 1$ by

$$\begin{aligned} \theta^* &= \text{acos}(1 - 2R_2) \qquad && (0 < \theta^* \leq \pi) \\ \phi &= 2\pi R_3 && (0 < \phi \leq 2\pi) \end{aligned}$$

ϕ is a relativistic invariant, and θ in the laboratory frame (Fig. 8) is given by

$$\tan\theta = \frac{1}{\gamma_1}\,\frac{\sin\theta^*}{\dfrac{\beta_1}{\beta_2^*} + \cos\theta^*}$$

β_2^* and momentum p_2 are given by

$$\begin{aligned} \gamma_2^* &= \frac{M_1^2 + M_2^2 - M_3^2}{2M_1M_2} \\ \gamma_2 &= \gamma_1\gamma_2^*\left(1 + \beta_1\beta_2^*\cos\theta^*\right) \\ \beta_2 &= \left(1 - \frac{1}{\gamma_2^2}\right)^{1/2} \\ p_2 &= M_2\sqrt{\gamma_2^2 - 1} \end{aligned}$$

Finally, θ and ϕ are transformed into the angles T_2 and P_2 in the **zgoubi** frame, and the relative momentum takes the value $D_2 = \dfrac{p_2}{q}\dfrac{1}{BORO}$ (where *BORO* is the reference rigidity, see *OBJET*), while the starting position of M_2 is the very location of the parent particle decay, (Y_1, Z_1, s_1).

The decay simulation by **zgoubi** satisfies the following procedures. In optical elements and field maps, after each integration step *XPAS*, the actual path length of the particle, $F(6, I)$, is compared to its limit path length s. If s is passed, then the particle is considered as having decayed at $F(6, I) - \dfrac{XPAS}{2}$, at a position obtained by a linear translation from the position at $F(6, I)$. Presumably, the smaller *XPAS*, the smaller the error on position and angles at the decay point.

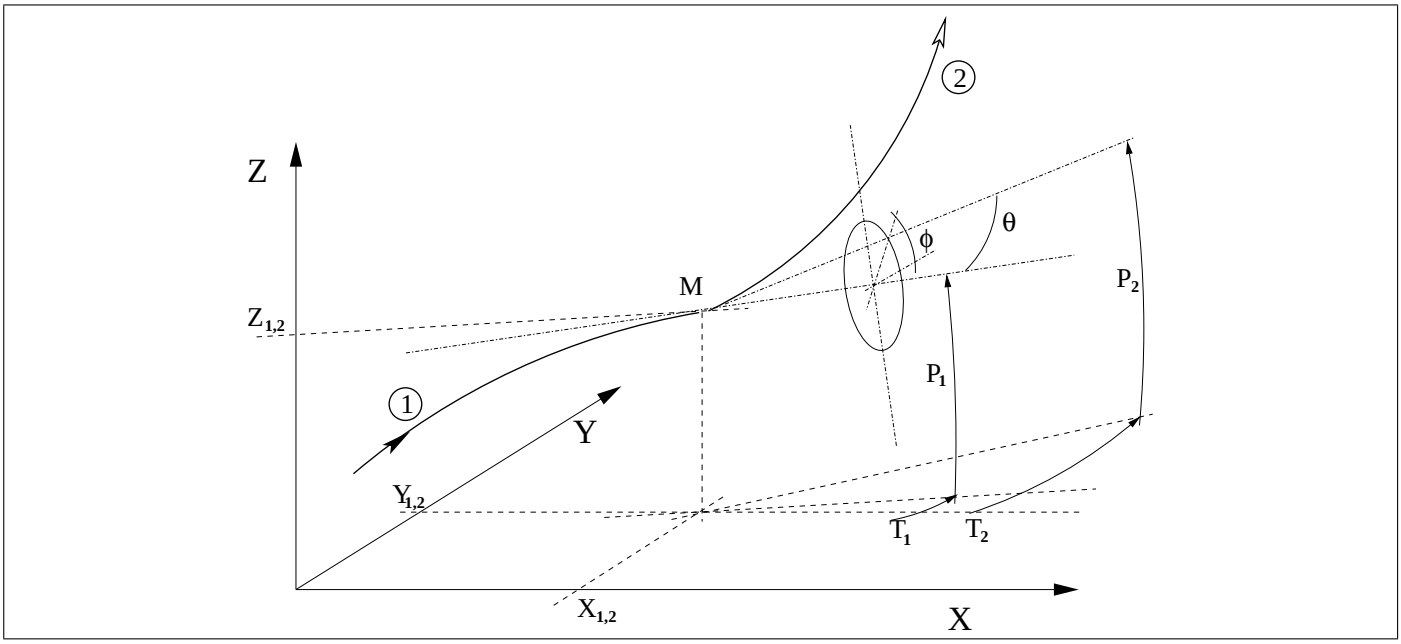


Figure 8: At position $M(X_1, Y_1, Z_1)$, particle 1 decays into 2 and 3 ; **zgoubi** then proceeds with the computation of the trajectory of 2, while 3 is discarded.
θ and ϕ are the scattering angles of particle 2 relative to the direction of the incoming particle 1 ; they transform to T_2 and P_2 in **zgoubi** frame.

In *ESL* and *CHANGREF*, $F(6, I)$ is compared to s at the end of the element. If the decay occurs inside the element, the particle is considered as having decayed at its actual limit path length s, thus its coordinates at s are recalculated by translation.

The limit path length of all particles ($I = 1$, *IMAX*) is stored in the array *FDES*$(6, I)$. For further statistical purposes (*e.g.*, use of *HISTO*) the daughter particle 2 is tagged with an $'S'$ standing for "secondary". When a particle decays, its coordinates D, Y, T, Z, P, s, *time* at the decay point are stored in *FDES*(J, I), $J = 1, 7$.

A note on negative drifts :
The use of negative drifts with *MCDESINT* is allowed and correct. For instance, negative drifts may occur in a structure for some of the particles when using *CHANGREF* (due to the Z-axis rotation or a negative *XCE)*, or when using *DRIFT* with *XL* < 0. Provision has been made to take it into account during the *MCDESINT* procedure, as follows.
If, due to a negative drift, a secondary particle reaches back the decay location of its parent particle, then the parent particle is "resurrected " with its original coordinates at that location, the secondary particle is discarded, and ray-tracing resumes in a regular way for the parent particle which is again allowed to decay, after the same path length. This procedure is made possible by prior storage of the coordinates of the parent particles (in array *FDES*(J, I)) each time a decay occurs.
Negative steps (*XPAS*< 0) in optical elements are not compatible with *MCDESINT*.

5.3.8 OPTICS : Write out optical functions. Log to zgoubi.OPTICS.out

OPTICS normally appears next to object definition, it normally works in conjunction with element label(s). *OPTICS* causes the transport and write out, in zgoubi.res, of the 6×6 beam matrix, following options *KOPT* and *'label'*, below.

IF *KOPT=0* : Off

IF *KOPT=1* : Will transport the optical functions with initial values as specified in *OBJET*, option *KOBJ=5.1*. Note that *OPTICS* assumes the first particle of the set of 11 as defined by *OBJET*, to be the reference particle for computation of the transport coefficients.

Note : In the case of a periodic structure, the initial coordinate values in *OBJET[KOBJ=5.1]* may be the periodic ones (as obtained, for instance, from a first run using *MATRIX[IFOC=11]*). Using *TWISS* keyword instead may be considered in that case.

A second argument, *'label'*, allows

- if *label = all* : printing out, into zgoubi.res, after all keywords of the zgoubi.dat structure,
- otherwise, printing out at all keyword featuring *LABEL* $\equiv$ *label* as a first label (see section 5.6.6, page 180, regarding the labelling of keywords).

PRINT : An optional third argument, *IMP=1*, or equivalently the command 'PRINT', will cause saving of the transported optical functions into file zgoubi.OPTICS.out.

About the source code :

The program `beamat` ensures the transport of the beam matrix along the zgoubi.dat sequence. `beamat` is called by `opticc`, itself called after each optical element, in the program `zgoubi`.

The starting beam matrix value is in the array `FI`, initialized by *OBJET[KOBJ=5.1]*, program `obj5`. The array `F0` contains the running beam matrix.

The program `beaimp` prints to zgoubi.res, whereas `optimp` prints to zgoubi.OPTICS.out. Both are called from `opticc`.

5.3.9 OPTIONS : Global or special options

OPTIONS allows on/off switching of various options.

Available, for now :

- “WRITE ON or OFF”
  Role : inhibit (most of) write statements to zgoubi.res.

Interest : “WRITE OFF” may allow substantial savings on CPU time. “WRITE ON” sets back to normal (the ’normal’ state of write outs may depend on the status of other commands, such as *REBELOTE*, on-going *FIT[2]*, etc.).

- “CONSTY ON or OFF”
  Role : force to constant *Y* and *Z* coordinates during stepwise integration.
  This allows for instance checking fields, or fabricating field maps from **zgoubi**’s analytical models of optical elements, as follows.

Concurrent use of with $IL = 1,\ 2,$ or 7 (*cf.* Sec. 5.6.5, p. 179) will allow logging $\vec{B}$ and/or $\vec{E}$ field(s) (and their derivatives) experienced across optical elements, in zgoubi.res, zgoubi.plt or zgoubi.impdev.out respectively.

As a consequence, forcing *Y* and *Z* to constant values, using “CONSTY ON”, allows checking of fields (and derivatives) at constant *Y* and *Z*, as determined by *OBJET*, across (individual or short series of) optical elements. This may be helpful when setting up input data in **zgoubi**. It can be for instance of great help in case of complex geometrical and field data inputs, as in *DIPOLES, FFAG*.

Initial coordinates are set in a very regular manner using *OBJET*. The Y coordinate will stand for the radius value in the case of optical elements defined in polar coordinates (*e.g.*, *DIPOLES*, polar field maps, etc.).

5.3.10 ORDRE : Taylor expansions order

The position $\vec{R}$ and velocity $\vec{u}$ of a particle are obtained from Taylor expansions as described in eq. (1.2.4). By default, these expansions are up to the fifth order derivative of $\vec{u}$,

$$\vec{R}_1 \approx \vec{R}_0 + \vec{u}\Delta s + ... + \vec{u}^{(5)}\,\frac{\Delta s^6}{6!}$$
$$\vec{u}_1 \approx \vec{u} + \vec{u}'\Delta s + \ldots + \vec{u}^{(5)}\,\frac{\Delta s^5}{5!}$$

which corresponds to fourth order derivatives of fields $\vec{B}$, eq. (1.2.8). and of $\vec{E}$, eq. (1.2.13).

However, third or higher order derivatives of fields may be zero in some optical elements, for instance in a sharp edge quadrupole. Also, in several elements, no more than first and second order field derivatives are implemented in the code. One may also wish to save on computation time by limiting the time-consuming calculation of lengthy (while possibly ineffective in terms of accuracy) Taylor expansions.

In that spirit, the purpose of *ORDRE*, option $IO = 2 - 5$, is to allow for expansions to the $\vec{u}^{(n)}$ term in eq. 1.2.4. Default functioning is $IO = 4$, stated in *FORTRAN* file `block.f`.

Note the following :
As concerns the optical elements

DECAPOLE, DODECAPO, EBMULT, ELMULT, MULTIPOL, OCTUPOLE, QUADRUPO, SEXTUPOL

field derivatives (see eq. 1.2.10 p. 21, eq. 1.2.15 p. 23,) have been installed in the code according to $\vec{u}^{(5)}$ Taylor development order ; it may not be as complete for other optical elements. In particular, in electric optical elements field derivatives (eq. 1.2.15) are usually provided to no more than second order, which justifies saving on computing time by means of *ORDRE*, so to avoid pushing Taylor expansions as high as $\vec{u}^{(5)}$.

NOTE : see also the option *IORDRE* in field map declarations (*DIPOLE-M, TOSCA*, etc.).

5.3.11 PARTICUL : Particle characteristics

Since **zgoubi** works using the rigidity, (*BORO*, as declared in *[MC]OBJET*), *PARTICUL* only needs be introduced (normally, following *[MC]OBJET* in the input data file zgoubi.dat) when the definition of some characteristics of the particles (mass, charge, gyromagnetic factor, life-time in the center of mass) is needed, as is the case when using the following procedures :

CAVITE	: mass, charge
MCDESINT	: mass, COM life-time
SPNTRK	: mass, gyromagnetic factor
SRLOSS	: mass, charge
SYNRAD	: mass, charge
Electric and Electro-Magnetic elements	: mass, charge

The declaration of *PARTICUL* must **precede** these keywords.

If *PARTICUL* is omitted, which is in general the case when ray-tracing ions in purely magnetic optical assemblies, then **zgoubi**, since it only knows the rigidity, will skip the computation of such quantities as time of flight.

5.3.12 REBELOTE : 'Do it again'

When *REBELOTE* is encountered in the input data file, the code execution jumps,
- either back to the beginning of the data file - the default behavior,
- or, if option *K=99.1* or *K=99.2*, back to a particular *LABEL*.

Then *NPASS-1* passes (from *LABEL* to *REBELOTE*) follow.
As to the last pass, number *NPASS+1*, there are two possibilities :
- either it also encompasses the whole *LABEL* to *REBELOTE* range,
- or, upon request (option *K=99.2*), execution may exit that final pass upstream of *REBELOTE*, at a location defined by a second dedicated *LABEL* placed between the first above mentioned *LABEL*, and *REBELOTE*. In both cases, following the end of this "multiple-pass" procedure, the execution continues from the keyword which follows *REBELOTE*, until *'END'* is encountered.

The two functionalities of *REBELOTE* are the following :

• *REBELOTE* can be used for Monte Carlo simulations when more than Max(*IMAX*) particles are to be tracked. Thus, when the following random procedures are used : *MCOBJET, OBJETA, MCDESINT, SPNTRK* (*KSO* = 5), their random seeds are not reset and independent statistics will add up.
This includes **Monte Carlo simulations**, in beam lines : normally $K = 0$. *NPASS* runs through the same structure, from *MCOBJET* to *REBELOTE* will follow, resulting in the calculation of $(1 + NPASS) * IMAX$ trajectories, with as many random initial coordinates.

• *REBELOTE* can be used for multi-turn ray-tracing in circular machines **circular machines** : normally $K = 99$ in that case. *NPASS* turns in the same structure will follow, resulting in the tracking of *IMAX* particles over $1 + NPASS$ turns. For the simulation of pulsed power supplies, synchrotron motion, and other Q-jump manipulation, see *SCALING*.
For instance, using option described *K=99.2* above, a full "injection line + ring + extraction line" installation can be simulated - kicker firing and other magnet ramping can be simulated using *SCALING*.
Using the double-*LABEL* method discussed above with option *K=99.2*, it is possible to encompass the ring between an injection line section (namely, with the element sequence of the latter extending from *OBJET* to the first *LABEL*), and an extraction line (its description will then follow *REBELOTE*), whereas the ring description extends from to the first *LABEL* to *REBELOTE*, with possible extraction, at the last pass, at the location of the second *LABEL*, located between the first one and *REBELOTE*,

In addition to what precedes, *REBELOTE* can change the value of arbitrary parameters in zgoubi.dat data list, using the forth argument *IOPT=1* (see page 270). *NPRM* tells the number of parameters to be changed. A series of *NPRM* lists of values, one list per parameter to be changed and each list with *NRBLT* data, tells the values to be taken by each one of these parameters, over the *NRBLT* passes of the *REBELOTE* process.

Output prints over *NPASS+1* passes might result in a prohibitively big zgoubi.res file. They may be switched on/off by means of the option *KWRIT*$= i.j$, with $i = 1/0$ respectively. The j flag commands printing pass number and some other information onto the video output, every 10^{j-1} turns if $j > 0$; output is switched off if $j = 0$.

REBELOTE also provides information : statistical calculations and related data regarding particle decay *(MCDESINT)*, spin tracking *(SPNTRK)*, stopped particles (*CHAMBR, COLLIMA*), etc.

COMBINING REBELOTE AND FIT[2]

The keyword *REBELOTE* can follow *FIT[2].* This allows executing again the fit procedure, after changing the value of some parameter(s) in zgoubi.dat using *REBELOTE* with option *IOPT=1.* That's the interest of the game : *REBELOTE* changes that (these) parameter(s), and then sends the zgoubi execution pointer back to the top of zgoubi.dat for a new *FIT[2]* run.

Conversely, *REBELOTE* may be followed by the keyword *FIT[2].* This allows for instance optimization of constraints that require multi-turn tracking (*e.g.*, optimized tune-jump quadrupole settings for beam acceleration through depolarizing resonances).

EXAMPLES

An example of the use of the *REBELOTE* procedure is given page 271.

An example of *FIT* preceding *REBELOTE* is given page 234.

RECOVERING FROM A CRASH

When accelerating a bunch, *REBELOTE* is used to ensure the multi-turn process. Now, if the job is stopped - due to computer crash or else - it is possible to resume the tracking from the latest records (in [b_]zboubi.fai storage file), for instance using *OBJET*, *KOBJ*= 3. *CAVITE* and *SCALING* aspects of the data file modifications necessary to ensure the recovery, in the case of pulsed magnets, are discussed respectively in the *SCALING* section, pp. 80, 82, an example of job resuming is given page 255.

5.3.13 RESET : Reset counters and flags

Resets counters involved in *CHAMBR*, *COLLIMA*, *HISTO* and *INTEG* procedures.

Switches off *CHAMBR*, *MCDESINT*, *SCALING* and *SPNTRK* options.

5.3.14 SCALING : Power supplies and R.F. function generator

SCALING acts as a function generator dedicated to varying fields in optical elements, potentials in electrostatic devices, possibly in correlation with RF parameters in *CAVITE.* It is normally intended to be declared right after the object definition.

Used in conjunction with *REBELOTE* for the simulation of multi-turn tracking, *SCALING* allows to change fields turn-by-turn in an arbitrary, user defined manner during an acceleration multi-turn. Fields (for instance main magnets') can as well be forced to follow a rigidity change caused by *CAVITE*, this can be obtained by simply using *NT=-1* as the first argument in *SCALING.*

The latter (*i.e.*, fields being correlated to *CAVITE*) holds in general, allowing as well the simulation of a recirculating linac (thus optical elements between cavities are scaled, following the rigidity increase), or multiple RF stations in a ring.

SCALING acts on families of elements, a family being designated by its name that coincides with the keyword of the corresponding element. For instance, declaring *MULTIPOL* as to be scaled will result in the same timing law being applied to all *MULTIPOL*s in zgoubi.dat optical structure data file. Subsets can be selected by labeling keywords in the data file (section 5.6.6, page 180) and adding the corresponding *LABEL*(s) in the *SCALING* declarations (9 *LABEL*s maximum - this can be changed in *MXFS.H*, parameter *MLF*). The family name of concern, as well as the scaling function for that family, are given as input data to the keyword *SCALING.* There is an upper limit to the number NF of families that can be declared as subject to a scaling law (this can be changed in the *FORTRAN* include file `MXFS.H`, parameter *MXF*).

A scaling law can be comprised of up to NT successive timings. A linear interpolation will determine the value of the scaling factor between successive timings.

If a family appears more than once in the *SCALING* list (*i.e.*, identical keyword and label list), the following will occur :

- if each instance addresses a different parameter in the optical elements concenred, both scaling rules are applied,

- if two (or more) instance address the same parameter in the optical elements, the last instance will prevail.

If a family addressed is a subset of a previously declared one, the *SCALING* rule will override the previous one, for that subset only.

An example of data formatting for the simulation of an acceleration cycle in a circular machine is given in the following.

SCALING		
1 5		Active. $NF = 5$ families of elements are concerned, as listed below :
QUADRUPO QFA QFB		- *QUADRUPO* keywords labeled 'QFA' and Quadrupoles labeled 'QFB'
2		$NT = 2$ timings
18131.E-3	24176.E-3	The field increases (linearly) from 18131E-3$*B_0$ to 24176E-3$*B_0$
1	6379	from turn 1 to turn 6379
MULTIPOL		- General rule. Precedes labeled cases if any.
2		$NT = 2$ timings
18131.E-3	24176.E-3	Fields increase from 18131E-3$*B_i$ to 24176E-3$*B_i$ ($\forall i = 1,\ 10$ poles)
1	6379	from turn 1 to turn 6379
MULTIPOL QDA QDB		- *MULTIPOL* keywords labeled 'QDA' and Multipoles labeled 'QDB'
2		$NT = 2$ timings
18131.E-3	24176.E-3	Fields increase from 18131E-3$*B_i$ to 24176E-3$*B_i$ ($\forall i = 1,\ 10$ poles)
1	6379	from turn 1 to turn 6379
BEND		- All *BEND* keywords, regardless of any *LABEL*
2		$NT = 2$ timings
18131.E-3	24176.E-3	As above
1	6379	
CAVITE		- Accelerating cavity
3		$NT = 3$ timings
1 1.22	1.33352	The synchronous rigidity $(B\rho)_s$ increases,
1 1200	6379	from $(B\rho)_{s_o}$ to 1.22 $*(B\rho)_{s_o}$ from turn 1 to 1200, and
		from 1.22 $*(B\rho)_{s_o}$ to 1.33352 $(B\rho)_{s_o}$ from turn 1200 to 6379

The timing is in unit of turns. In this example, TIMING = 1 to 6379 (turns). Therefore, at turn number $N,\ B$ and B_i are updated in the following way. Let *SCALE*(*TIMING* $= N$) be the updating scale factor

$$\text{SCALE}(N) = 18.131\frac{24.176 - 18.131}{1 + 6379 - 1}(N - 1)$$

and then

$$B(N) = \textit{SCALE}(N)B_0$$
$$B_i(N) = \textit{SCALE}(N)B_{i0}$$

The RF frequency is computed using

$$f_{RF} = \frac{hc}{\mathcal{L}}\frac{q(B\rho)_s}{(q^2(B\rho)_s^2 + (Mc^2)^2)^{1/2}}$$

where the rigidity is updated in the following way. Let $(B\rho)_{s_o}$ be the initial rigidity (namely, $(B\rho)_{s_o} =$ *BORO* as defined in the keyword *OBJET* for instance). Then, at turn number N,

$$\text{if } 1 \le N \le 1200 \text{ then, } \textit{SCALE}(N) = 1 + \frac{1.22 - 1}{1 + 1200 - 1}(N - 1)$$
$$\text{if } 1200 \le N \le 6379 \text{ then, } \textit{SCALE}(N) = 1.22 + \frac{1.33352 - 1.22}{1 + 6379 - 1200}(N - 1200)$$

and then,

$$(B\rho)_s(N) = \textit{SCALE}(N) \cdot (B\rho)_{s_o}$$

from which value the calculations of $f_{RF}(N)$ follow.

NT can take negative values, then acting as an option switch (rather than giving number of timings), as follows :

• $NT = -1$: this is convenient for synchrotron acceleration. In this case the next two lines both contain a single data (as for $NT = 1$), respectively the starting scaling factor value, and 1. The current field scaling factor will then be updated from the energy kick by the cavity if for instance *CAVITE/IOPT=2* is used, namely,

$$\text{SCALE}(N) = \text{SCALE}(N-1) * \frac{B\rho(N)}{B\rho(N-1)}$$

Note that in the case of the previous example (assuming, though, that the ramping of the magnetic fields strictly follows the *CAVITE* kick, which was not strictly the case), the *SCALING* argument list can be simplified, including by suppressing *'CAVITE'* from the list, as follows :

SCALING	- Scaling
1 3	Active. $NF = 3$ families of elements are concerned, as listed below
QUADRUPO QFA QFB	- Quadrupoles labeled 'QFA' and Quadrupoles labeled 'QFB'
-1	
18131.E-3	
1	
MULTIPOL QDA QDB	- Multipoles labeled 'QDA' and Multipoles labeled 'QDB'
-1	
18131.E-3	
1	
BEND	- All *BEND*s (regardless of any *LABEL*)
-1	
18131.E-3	
1	

• $NT = -2$: this is convenient for reading an RF law for *CAVITE* from an external data file, including usage for acceleration in fixed field accelerators.

• $NT = 1.10$: allows taking the scaling law from an external data file, as in the following example :

MULTIPOL COH1	
1.10	
./Csnk3D/bump_centered.scal	File name
1 2	Column numbers in the file : col. 2 gives the scaling factor at rigidity given by col. 1.

Notes :

1. In causing, via *CAVITE*, a change of the synchronous rigidity, *SCALING* causes a change of the reference rigidity, following (see *CAVITE*)

$$B\rho_{ref} = \textit{BORO} \longrightarrow B\rho_{ref} = \textit{BORO} + \delta B\rho_s$$

2. It may happen that some optical elements won't scale, for source code development or updating reasons. This should be paid attention to by preliminary simulation tests.

RECOVERING FROM A CRASH

When accelerating a bunch in a pulsed ring, *SCALING* is used to ramp magnetic fields, so following the rigidity increase by *CAVITE*. *CAVITE* may or may not be part of the *SCALING* argument list, see the two previous examples. Now, if the job is stopped - due to computer crash or else - it is possible to resume the tracking from the latest records (in [b_]zboubi.fai storage file), for instance using *OBJET*, *KOBJ*= 3. In the presence of acceleration using *CAVITE* however, since there is no reference particle in **zgoubi**, *CAVITE* needs be told the reference rigidity at the corresponding time/turn where tracking is resumed. This is done using *SCALING*, see example page 255. This has the effect of updating the refrence rigidity $B\rho_{ref}$ to its correct value (see Note 1 above), the one it had when the job stopped.

5.3.15 SPACECHARG : Space charge

The keyword *SPACECHARG* enables (or disables) the calculation of space charge effects. It must be preceded by *PARTICUL* for definition of mass and charge values, as they enter in the definition of space charge parameters.

SPACECHARG is supposed to appear a first time at the location where space charge effects should start being taken into account, with the first data *LMNT* set to 'all' or a keyword.

SPACECHARG is supposed to appear a second time at the location where calculations should stop, with *LMNT* set to 'none'. It results in summary outputs in zgoubi.res.

Occurence of *PRINT* amongst *SPACECHARG* arguments results in storage of computational data and other relevant informations in zgoubi.SPACECHARG.out, as space charge computation proceeds.

5.3.16 SPNTRK : Spin tracking

The keyword *SPNTRK* allows switching spin tracking on (index *KSO*=1) or off (*KSO*=0), or resuming (index *KSO*=-1, following an occurence *KSO*=0). It also permits the attribution of an initial spin to each one of the *IMAX* particles of the beam, following a distribution that depends on the option index *KSO*. It must be preceded by *PARTICUL* for the definition of mass and gyromagnetic factor.

KSO = 1 (respectively 2, 3) : the *IMAX* particles defined with *[MC]OBJET* are given a longitudinal (1,0,0) spin component (respectively transverse horizontal (0,1,0), vertical (0,0,1)).

KSO = 4 : initial spin components are entered explicitly for each one of the *IMAX* particles of the beam.

KSO = 4.1 : three initial spin components $S_X,\ S_Y,\ S_Z$ are entered explicitly just once, they are then assigned to each one of the *IMAX* particles of the beam.

KSO = 4.2 : *under development.* Read spin components from a file, file name has to be provided. (that file at present can be zgoubi.res : KSO=4.2 looks for key "(deg) (deg)" that appears at start of the spin coordinates list under the keyword *SPNPRT*).

KSO = 5 : random generation of *IMAX* initial spin conditions as described in Fig. 9. Given a mean polarization axis (S) defined by its angles T_0 and P_0, and a cone of angle A with respect to this axis, the *IMAX* spins are sorted randomly in a Gaussian distribution

$$p(a) = \exp\left[-\frac{(A-a)^2}{2\delta A^2}\right] / \delta A\sqrt{2\pi}$$

and within a cylindrical uniform distribution around the (S) axis. Examples of simple distributions available by this mean are given in Fig. 10.

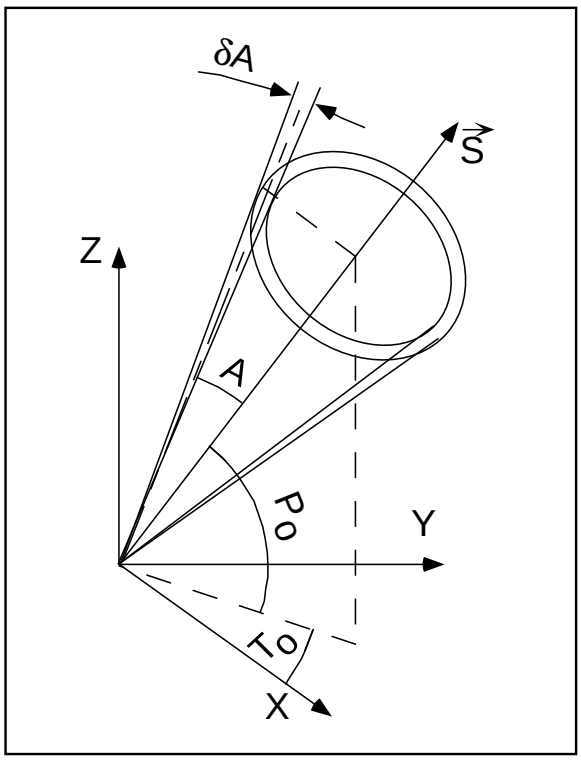


Figure 9: Spin distribution as obtained with option *KSO* = 5.
The spins are distributed within an annular strip δA (standard deviation) at an angle A with respect to the axis of mean polarization ($\vec{S}$) defined by T_0 and P_0.

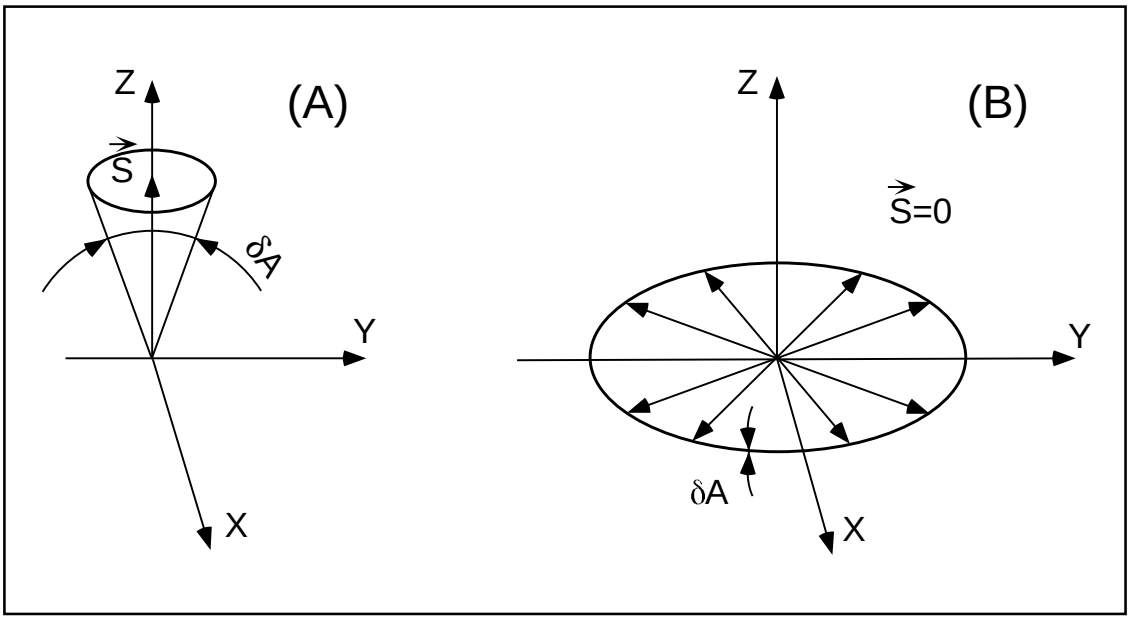


Figure 10: Examples of the use of *KSO* =5.
A : Gaussian distribution around a mean vertical polarization axis, obtained with T_0 = arbitrary, $P_0 = \pi/2$, $A = 0$ and $\delta A \neq 0$.
B : Isotropic distribution in the median plane, obtained with $P_0 = \pm\pi/2$, $A = \pi/2$, and $\delta A = 0$.

5.3.17 SRLOSS : Synchrotron radiation energy loss [19]

The keyword *SRLOSS* allows activating or stopping (option $KSR = 1, 0$ respectively) stepwise tracking of energy loss by stochastic emission of photons in magnetic fields, following the method described in section 3.1.

It can be chosen to allow radiation in the sole dipole fields, or in all types of fields regardless of their multipole composition. It can also be chosen to allow for the radiation induced transverse kick.

SRLOSS must be preceded by *PARTICUL* for defining mass and charge values as they enter in the definition of SR parameters.

Statistics on SR parameters are computed and updated while tracking, the results of which can be obtained by means of the keyword *SRPRNT*.

5.3.18 SYNRAD : Synchrotron radiation spectral-angular densities

The keyword *SYNRAD* enables (or disables) the calculation of synchrotron radiation (SR) electric field and spectral angular energy density. It must be preceded by *PARTICUL* for defining mass and charge values, as they enter in the definition of SR parameters.

SYNRAD is supposed to appear a first time at the location where SR should start being taken into account, with the first data *KSR* set to 1. It results in on-line storage of the electric field vector and other relevant quantities in zgoubi.sre, as step by step integration proceeds. The observer position (XO, YO, ZO) is specified next to *KSR*.

Data stored in zgoubi.sre :

(ELx, ELy, ELz) : electric field vector $\vec{\mathcal{E}}$ (eq. 3.2.1)
$(btx, bty, btz) = \vec{\beta} = \frac{1}{c} \times$ particle velocity
$(gx, gy, gz) = \frac{d\vec{\beta}}{dt} =$ particle acceleration (eq. 3.2.3)
$\Delta\tau =$ observer time increment (eq. 3.2.2)
$t' = \tau - r(t')/c =$ retarded (particle) time
(rtx, rty, rtz) : $\vec{R}(t)$, particle to observer vector (eq. 3.2.4)
$(x, y, z) =$ particle coordinates
$\Delta s =$ step size in the magnet (fig. 2)
$NS =$ step number
$I =$ particle number
$LET(I) =$ tagging letter
IEX$(I) =$ stop flag (see section 5.6.11)

SYNRAD is supposed to appear a second time at the location where SR calculations should stop, with *KSR* set to 2. It results in the output of the angular energy density $\int_{\nu_1}^{\nu_2} \partial^3 W/\partial\phi\,\partial\psi\,\partial\nu$ (eq. 3.2.11) as calculated from the Fourier transform of the electric field (eq. 3.2.11). The spectral range of interest and frequency sampling (ν_1, ν_2, N) are specified next to *KSR*.

5.3.19 SYSTEM : System call

The keyword *SYSTEM* allows one or a series of system calls. It can appear anywhere, an arbitrary number of times, in the zgoubi.dat data list. It is effective at the very location where it appears.

SYSTEM keyword is followed by the list of the desired system commands. That can be saving zgoubi output files, calling again **zgoubi** at the end of a run so allowing dependent consecutive jobs, etc.

Examples of the use of *SYSTEM* are given in pp. 234, 286.

5.4 Optical Elements and Related Numerical Procedures

AGSMM : AGS main magnet

The AGS main magnet is a combined function dipole with straight axis (lines of constant field are straight lines).
The field computation routines for *AGSMM* are the same as for *MULTIPOL* (details in section 1.3.7, page 27), however *AGSMM* has the following four particularities :

- There are only three multipole components present in *AGSMM* : dipole, quadrupole and sextupole.
- The dipole field B_0 is drawn from the reference rigidity, $B\rho_{ref}$, and follows the latter so to preserve $\rho = B\rho_{ref}/B_0$ and the orbit deviation L/ρ. In particular,
  - in the absence of acceleration, $B\rho_{ref} \equiv BORO$, with $BORO$ the quantity appearing in the object definition using *[MC]OBJET*,
  - in presence of acceleration using *CAVITE*, $B\rho_{ref}$ is changed to $BORO \times D_{ref}$ at each passage in the cavity, with D_{ref} the relative synchronous momentum increase, a quantity that **zgoubi** updates at cavity traversal.
- The field indices, quadrupole $K1$ and sextupole $K2$, are derived from the reference rigidity, $B\rho_{ref}$, via momentum-dependent polynomials, taken from Ref. [35], and following in that the methods found in the MAD model of the AGS [36].
- The AGS main dipole has back-leg windings, used for instance for injection and extraction orbit bumps. The number of winding turns and the number of ampere-turns are part of the data in the input data list. The intensity in the windings is accounted for in the conversion from total ampere-turns in the magnet to momentum and then to magnetic field.

Note : A consequence of items 2 and 3 is that no field value is required in defining the AGS main magnets in the zgoubi.dat input data list.

AGSQUAD : AGS quadrupole

The AGS quadrupoles are regular quadrupoles. The simulation of *AGSQUAD* uses the same field modelling as *MULTIPOL*, section 1.3.7, page 27. However amperes are provided as input to *AGSQUAD* rather than fields, the reason being that some of the AGS quadrupoles have two superimposed coil circuits, with separate power supplies. It has been dealt with this particularity by allowing for an additional set of quadrupole data in *AGSQUAD*, compared to *MULTIPOL*.

The field in *AGSQUAD* is computed using transfer functions, from the specified ampere-turns in the coils to magnetic field at pole-tip, that account for the non-linearity of the magnetic permeability [36].

AIMANT : Generation of dipole mid-plane 2-D map, polar frame

The keyword *AIMANT* provides an automatic generation of a dipole median plane field map in polar coordinates.

A more recent and improved version will be found in *DIPOLE-M*. In addition, a similar modelling, that however skips the stage of an intermediate mid-plane field map, can be found in *DIPOLE[S].*

The extent of the map is defined by the following parameters, as shown in Figs. 11A and 11B,

AT	:	total angular aperture
RM	:	mean radius used for the positioning of field boundaries
RMIN, RMAX	:	minimum and maximum radial boundaries of the map

The 2 or 3 effective field boundaries (EFB) inside the map are defined from geometric boundaries, the shape and position of which are determined by the following parameters,

ACENT	:	arbitrary angle, used for the positioning of the EFBs.
ω	:	azimuth of an EFB with respect to *ACENT*
θ	:	angle of a boundary with respect to its azimuth (wedge angle)
R_1, R_2	:	radius of curvature of an EFB
U_1, U_2	:	extent of the linear part of the EFB.

At any node of the map mesh, the value of the Z component of the field is calculated as

$$B_Z = \mathcal{F}(R,\theta) * B_0 * \left(1 + N * \left(\frac{R-RM}{RM}\right) + B * \left(\frac{R-RM}{RM}\right)^2 + G * \left(\frac{R-RM}{RM}\right)^3\right) \qquad (5.4.1)$$

where N, B and G are respectively the first, second and third order field indices and $\mathcal{F}(R,\theta)$ is the fringe field coefficient (it determines the "flutter" in periodic structures).

Calculation of the Fringe Field Coefficient

With each EFB a realistic extent of the fringe field, λ, is associated (Figs. 11A and 11B), and a fringe field coefficient F is calculated. In the following λ stands for either λ_E (Entrance), λ_S (Exit) or λ_L (Lateral EFB).
If a node of the map mesh is at a distance of the EFB larger than λ, then $F = 0$ outside the field map and $F = 1$ inside. If a node is inside the fringe field zone, then F is calculated as follows.

Two options are available, for the calculation of F, depending on the value of ξ.

If $\xi \geq 0$, F is a second order type fringe field (Fig. 12) given by

$$F = \frac{1}{2}\,\frac{(\lambda - s)^2}{\lambda^2 - \xi^2} \quad \text{if } \xi \leq s \leq \lambda \qquad (5.4.2)$$

$$F = 1 - \frac{1}{2}\,\frac{(\lambda - s)^2}{\lambda^2 - \xi^2} \quad \text{if } -\lambda \leq s \leq -\xi \qquad (5.4.3)$$

where s is the distance to the EFB, and

$$F = \frac{1}{2} + \frac{s}{\lambda + \xi} \quad \text{if } 0 \leq s \leq \xi \qquad (5.4.4)$$

$$F = \frac{1}{2} - \frac{s}{\lambda + \xi} \quad \text{if } -\xi \leq s \leq 0 \qquad (5.4.5)$$

This simple model allows a rapid calculation of the fringe field, but may lead to erratic behavior of the field when extrapolating out of the median plane, due to the discontinuity of d^2B/ds^2, at $s = \pm\xi$ and $s = \pm\lambda$. For better accuracy it is advised to use the next option.

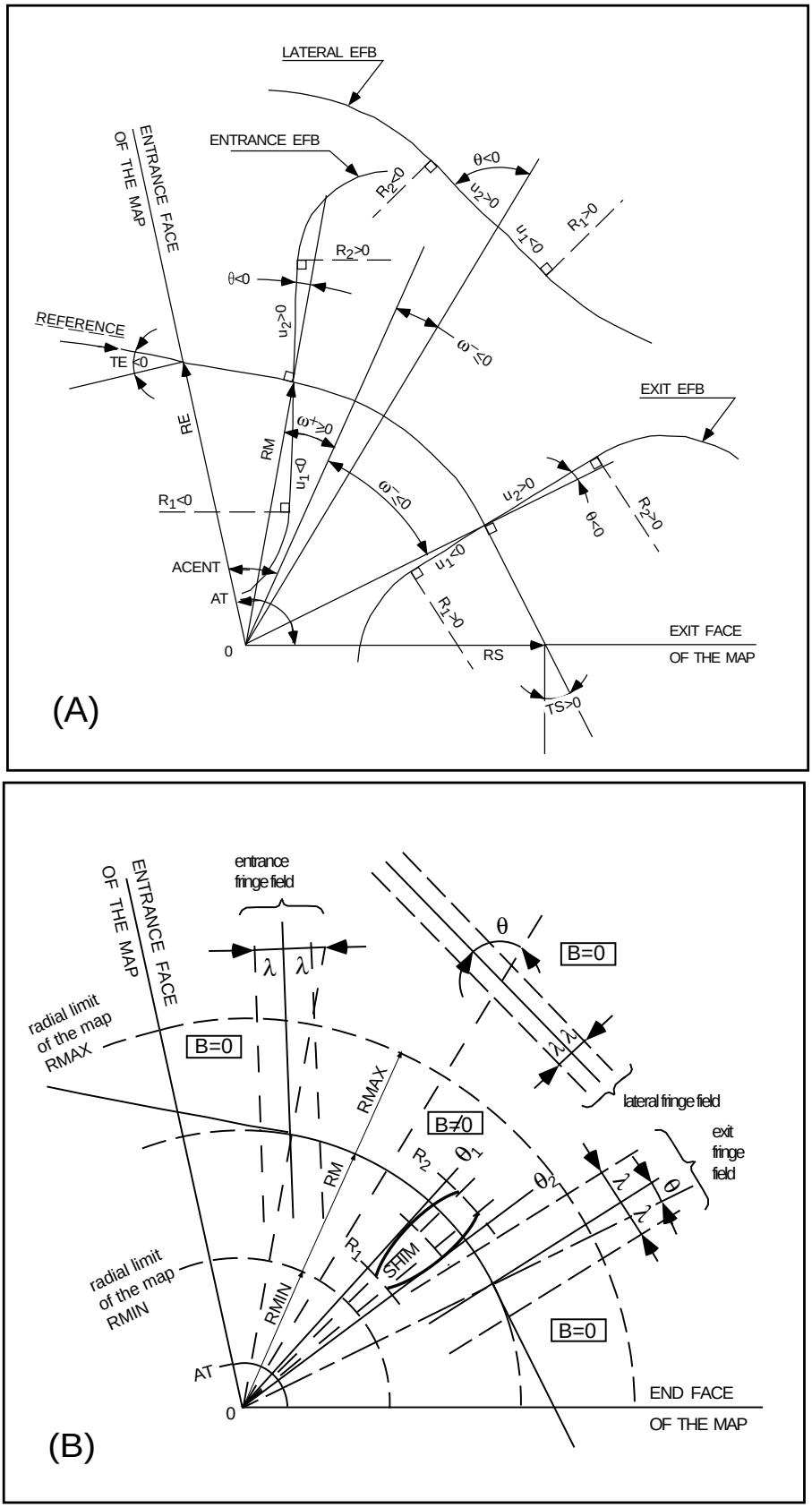


Figure 11: A : Parameters used to define the field map and geometrical boundaries.
B : Parameters used to define the field map and fringe fields.

If $\xi = -\mathbf{1}$, F is an exponential type fringe field (Fig. 12) given by [37]

$$F = \frac{1}{1 + \exp P(s)} \tag{5.4.6}$$

where s is the distance to the EFB, and

$$P(s) = C_0 + C_1 \left(\frac{s}{\lambda}\right) + C_2 \left(\frac{s}{\lambda}\right)^2 + C_3 \left(\frac{s}{\lambda}\right)^3 + C_4 \left(\frac{s}{\lambda}\right)^4 + C_5 \left(\frac{s}{\lambda}\right)^5 \tag{5.4.7}$$

The values of the coefficients C_0 to C_5 should be such that the derivatives of B_Z with respect to s be negligible at $s = \pm\lambda$, so as not to perturb the extrapolation of $\vec{B}$ out of the median plane.
It is also possible to simulate a shift of the EFB, by giving a non zero value to the parameter *shift*. s is then changed to $s-$ *shift* in the previous equation. This allows small variations of the total magnetic length.
Let F_E (respectively F_S, F_L) be the fringe field coefficient attached to the entrance (respectively exit, lateral) EFB following the equations above. At any node of the map mesh, the resulting value of the fringe field coefficient (eq. 5.4.1) is (Fig. 13)

$$\mathcal{F}(R, \theta) = F_E * F_S * F_L$$

In particular, $F_L \equiv 1$ if no lateral EFB is requested.

The Mesh of the Field Map

The magnetic field is calculated at the nodes of a mesh with polar coordinates, in the median plane. The radial step is given by

$$\delta R = \frac{\textit{RMAX - RMIN}}{\textit{IRMAX} - 1}$$

and the angular step by

$$\delta\theta = \frac{AT}{\textit{IAMAX} - 1}$$

where, *RMIN* and *RMAX* are the lower and upper radial limits of the field map, and AT is its total angular aperture (Fig. 11B). *IRMAX* and *IAMAX* are the total number of nodes in the radial and angular directions.

Simulating Field Defects and Shims

Once the initial map is calculated, it is possible to perturb it by means of the parameter *NBS*, so as to simulate field defects or shims.

If $\mathbf{NBS} = -\mathbf{2}$, the map is globally modified by a perturbation proportional to $R - R_0$, where R_0 is an arbitrary radius, with an amplitude $\Delta B_Z / B_0$, so that B_Z at the nodes of the mesh is replaced by

$$B_Z * \left(1 + \frac{\Delta B_Z}{B_0} \frac{R - R_0}{\textit{RMAX} - \textit{RMIN}}\right)$$

If $\mathbf{NBS} = -\mathbf{1}$, the perturbation is proportional to $\theta - \theta_0$, and B_Z is replaced by

$$B_Z * \left(1 + \frac{\Delta B_Z}{B_0} \frac{\theta - \theta_0}{AT}\right)$$

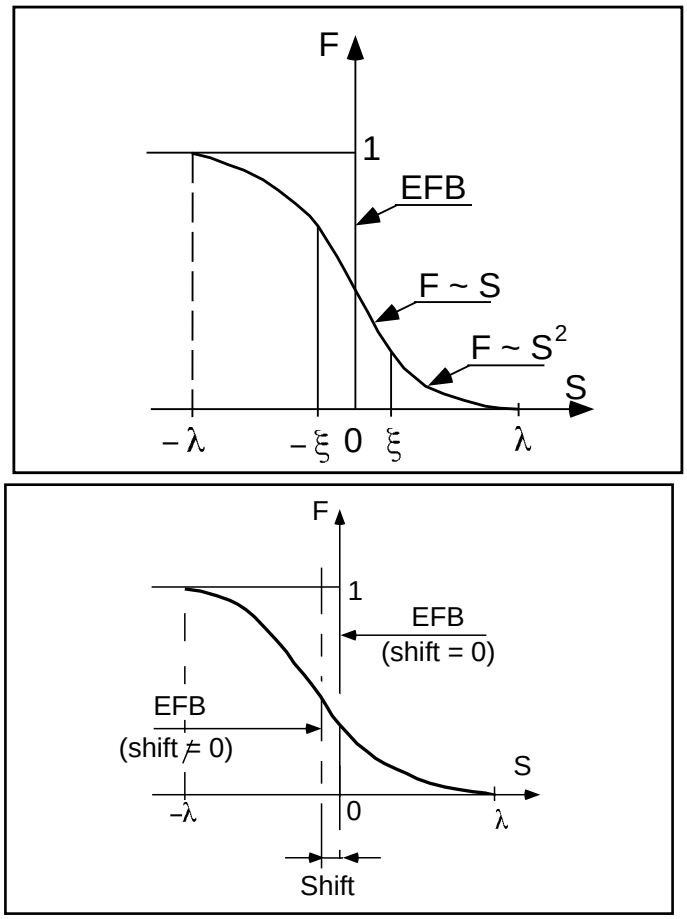


Figure 12: Second order type fringe field (upper plot) and exponential type fringe field (lower plot).

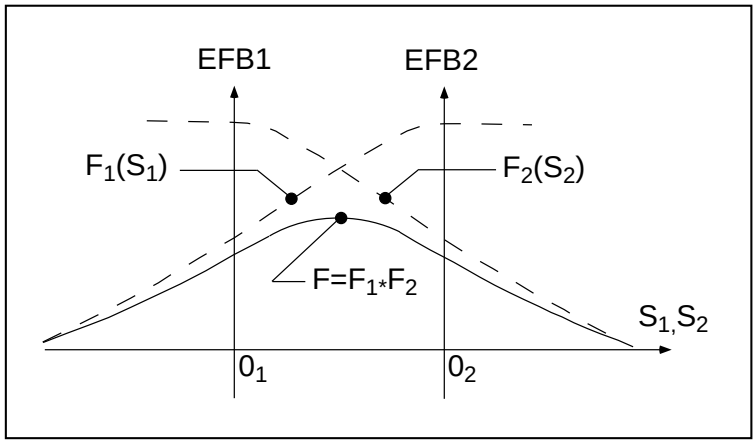


Figure 13: Effective value of $\mathcal{F}(R,\theta)$ for overlapping fringe fields F_1 and F_2 centered at O_1 and O_2.

If NBS $\geq$ 1, then *NBS* shims are introduced at positions $\frac{R_1+R_2}{2}$, $\frac{\theta_1+\theta_2}{2}$ (Fig. 14) [38]
The initial field map is modified by shims with second order profiles given by

$$\theta = \left(\gamma + \frac{\alpha}{\mu}\right) \beta \frac{X^2}{\rho^2}$$

where X is shown in Fig. 14, $\rho = \frac{R_1+R_2}{2}$ is the central radius, α and γ are the angular limits of the shim, β and μ are parameters.
At each shim, the value of B_Z at any node of the initial map is replaced by

$$B_Z * \left(1 + F\theta * FR * \frac{\Delta B_Z}{B_0}\right)$$

where $F\theta = 0$ or $FR = 0$ outside the shim, and $F\theta = 1$ and $FR = 1$ inside.

Extrapolation Off Median Plane

The vertical field $\vec{B}$ and its derivatives in the median plane are calculated by means of a second or fourth order polynomial interpolation, depending on the value of the parameter *IORDRE* (*IORDRE*=2, 25 or 4, see section 1.4.2). The transformation from polar to Cartesian coordinates is performed following eqs. (1.4.9 or 1.4.10). Extrapolation off median plane is then performed by means of Taylor expansions following the procedure described in section 1.3.3.

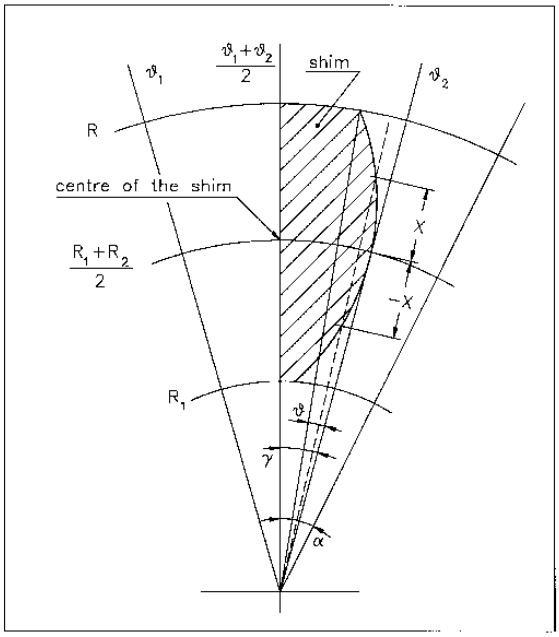


Figure 14: A second order profile shim. The shim is centered at $\frac{(R_1+R_2)}{2}$ and $\frac{(\theta_1+\theta_2)}{2}$.

AUTOREF : Transport beam into a new reference frame

AUTOREF positions the new reference frame following four different options (apart from $\mathbf{I = 0}$ which is just an "off" switch) :

If $\mathbf{I = 1}$, *AUTOREF* is equivalent to

$$\textit{CHANGREF}[XCE = 0, YCE = Y(1), ALE = T(1)]$$

so that the new reference frame is at the exit of the last element, with particle 1 at the origin with its horizontal angle set to $T = 0$.

If $\mathbf{I = 2}$, it is equivalent to

$$\textit{CHANGREF}[XW, YW, T(1)]$$

so that the new reference frame is at the position (XW, YW) of the waist (calculated automatically in the same way as for *IMAGE*) of the three rays number 1, 4 and 5 (compatible for instance with *OBJET*, *KOBJ* = 5, 6, together with the use of *MATRIX*) while $T(1)$, the horizontal angle of particle number $I1$, is set to zero.

If $\mathbf{I = 3}$, it is equivalent to

$$\textit{CHANGREF}[XW, YW, T(I1)]$$

so that the new reference frame is at the position (XW, YW) of the waist (calculated automatically in the same way as for *IMAGE*) of the three rays number I1, I2 and I3 specified as data, while $T(I1)$ is set to zero.

If $\mathbf{I = 4}$: new horizontal beam centroid positionning *XCE, YCE, ALE* is provided. The beam is moved by XCE and then centered on *YCE, ALE.*

If $\mathbf{I = 4.1}$: new beam centroid positionning *XCE, YCE, ALE, DCE, TIME* is provided. The beam is moved by *XCE* and then centered on *YCE, ALE.* In addition, the beam is centered on a new relative momentum *DCE* and new timing value *TIME.*

If $\mathbf{I = 4.2}$: same as 4.1, except that particles all have their timing set to *TIME.*

If $\mathbf{I = 5}$: new vertical beam centroid positionning *ZCE, PLE* (position, angle) is provided. The beam is centered on vertical position and angle *ZCE, PLE.*

BEAMBEAM : Beam-beam lens

BEAMBEAM is a beam-beam lens simulation, a point transform [39].

Upon option using *SPNTRK*, *BEAMBEAM* will include spin kicks, after modelling as described in Ref. [40].

BEND : Bending magnet, Cartesian frame

BEND is one of several keywords available for the simulation of dipole magnets. It presents the interest of easy handling, and is well adapted for the simulation of synchrotron dipoles and such other regular dipoles as sector magnets with wedge angles.

The field in *BEND* is defined in a Cartesian coordinate frame (unlike for instance *DIPOLE[S]* that uses a polar frame). As a consequence, having particle coordinates at entrance or exit of the magnet referring to the curved main direction of motion may require using *KPOS*, in particular *KPOS=3* (in a circular machine cell for instance, see section 5.6.8, p. 181).

The dipole simulation accounts for the magnet geometrical length *XL*, for a possible skew angle (X-rotation, useful for obtaining vertical deviation magnet), and for the field $B1$ such that in absence of fringe field the deviation θ satisfies $XL = 2\dfrac{BORO}{B1}\sin\frac{\theta}{2}$.

Then follows the description of the entrance and exit EFBs and fringe fields. The wedge angles W_E (entrance) and W_S (exit) are defined with respect to the sector angle, with the signs as described in Fig. 15. Within a distance $\pm X_E(\pm X_S)$ on both sides of the entrance (exit) EFB, the fringe field model is used (same as for *QUADRUPO*, Fig. 36, p. 148) ; elsewhere, the field is supposed to be uniform.

If λ_E (resp. λ_S) is zero sharp edge field model is assumed at entrance (resp. exit) of the magnet and X_E (resp. X_S) is forced to zero. In this case, the wedge angle vertical first order focusing effect (if $\vec{B}1$ is non zero) is simulated at magnet entrance and exit by a kick $P_2 = P_1 - Z_1\tan(\epsilon/\rho)$ applied to each particle (P_1, P_2 are the vertical angles upstream and downstream the EFB, Z_1 the vertical particle position at the EFB, ρ the local horizontal bending radius and ϵ the wedge angle experienced by the particle ; ϵ depends on the horizontal angle T).

Magnet (mis-)alignment is assured by *KPOS*. *KPOS* also allows some degrees of automatic alignment useful for periodic structures (section 5.6.8).

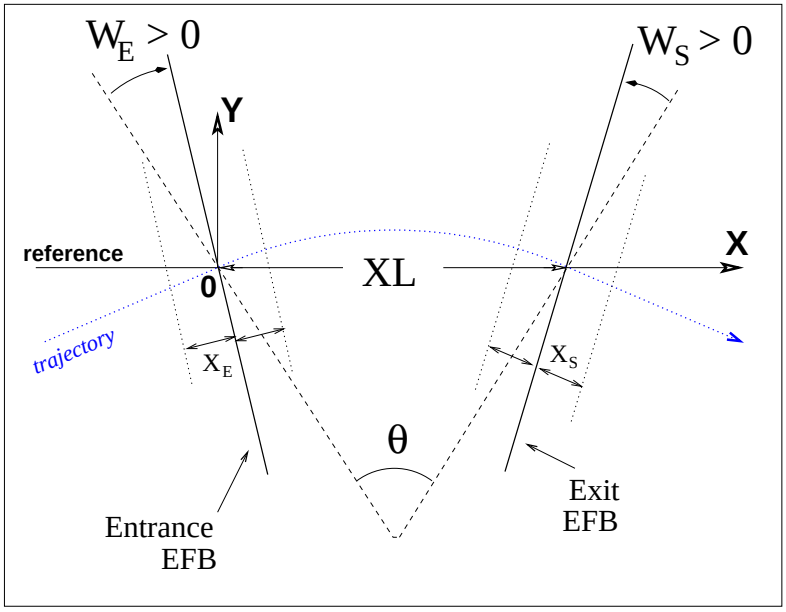


Figure 15: Geometry and parameters of *BEND* : *XL* = length, θ = deviation, W_E, W_S are the entrance and exit wedge angles. The motion is computed in the Cartesian frame (O, X, Y, Z)

From matrice-style code modeling, to zgoubi : Fig. 16 illustrates the conversion of matrix method style of data where the magnet is defined by its length and deviation angle (see MAD input below, using 'SBEND'), to **zgoubi** input data using *BEND* (in this particular case of an illustration of a rectangular magnet, *MULTIPOL* could be used, as well).

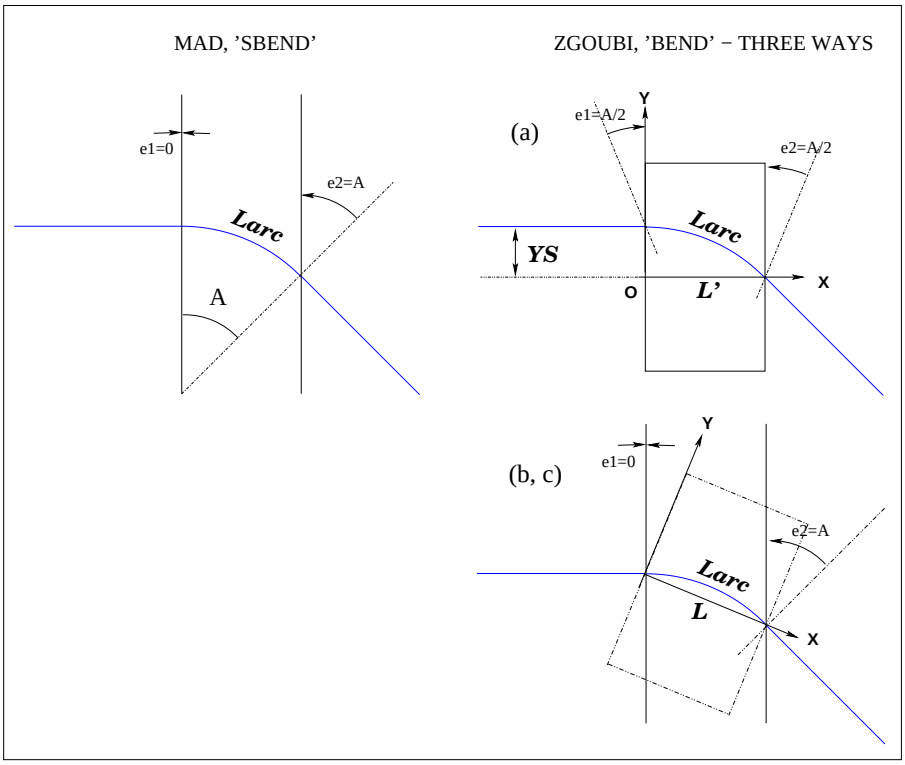


Figure 16: Simulating a matrix method style of rectangular magnet (*e.g.*, MAD's 'SBEND') by means of *BEND*, in three different ways, see text. In case (a), the straight length in **zgoubi**'s *BEND* is $L' = \rho \sin(A)$, whereas in cases (b, c) it is $L = 2\rho \sin(A/2)$.

• MAD style 'BEND' definition :

```
L = 1.
A = 0.1
Larc = L*(A/2.)/ sin(A/2.)
e1 = 0.
e2 = A
B: SBEND,  L=Larc, ANGLE=A, e1 = e1, e2 = e2
```

• **zgoubi**, case (a) :

```
(a) YS=-rho*(1-cos(A)) translated, non-rotated BEND,
    followed by A Z-rotation of frame

 'CHANGREF'
YS  -4.997916927068
 'BEND'
1
99.8750260394 0.   1.
0. 0. 0.05         ! e1=A/2
0 0. 0. 0. 0. 0. 0.
0. 0. 0.05         ! e2=A/2
0 0. 0. 0. 0. 0. 0.
.1
1 0. 0.  0.
 'CHANGREF'
ZR  -5.72957795131
```

• **zgoubi**, case (b) or (c) using *KPOS=1* or *3* :

```
(b) A/2 Z-rotated BEND followed by A/2 Z-rotation of frame

 'CHANGREF'
ZR  -2.86478897565
 'BEND'
1
100. 0.   1.
0. 0. 0.          ! e1=0
0 0. 0. 0. 0. 0. 0.
0. 0. 0.1         ! e2=A
0 0. 0. 0. 0. 0. 0.
.1
1 0. 0.  0.
 'CHANGREF'
ZR  -2.86478897565



(c) same as (b), using KPOS=3 instead

 'BEND'
1
100. 0.   1.
0. 0. 0.          ! e1=0
0 0. 0. 0. 0. 0. 0.
0. 0. 0.1         ! e2=A
0 0. 0. 0. 0. 0. 0.
.1
3 0. 0.  -0.05
```

Negative bend, vertical bend, tricks : Use *YMY* for the former, *TRAROT* with a $\pm\pi$ X-rotation for the latter. The two can be combined, so that a vertical negative bend can be represented by the sequence *TRAROT[π], YMY, BEND[B>0], YMY, TRAROT[$-\pi$]*, with positionning methods for *BEND* as discussed above (Fig. 16) still applying.

BREVOL : 1-D uniform mesh magnetic field map

BREVOL reads a 1-D axial field map from a storage data file, whose content must match the following *FORTRAN* reading sequence (possible FORMAT updates are to be found in `fmapw.f`).

```
      OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
      DO 1 I = 1, IX
        IF (BINARY) THEN
              READ(NL) X(I), BX(I)
        ELSE
              READ(NL,*) X(I), BX(I)
        ENDIF
1     CONTINUE
```

where *IX* is the number of nodes along the (symmetry) X-axis, $X(I)$ their coordinates, and *BX*(I) are the values of the X component of the field. *BX* is normalized with *BNORM* factor prior to ray-tracing, as well X is normalized with the coefficient *XNORM* (useful to convert to centimeters, the working units in **zgoubi**). For binary files, *FNAME* must begin with 'B_' or 'b_', a flag 'BINARY' will thus be set to '.TRUE.' by the *FORTRAN*.

X-cylindrical symmetry is assumed, resulting in BY and BZ taken to be zero on axis. $\vec{B}(X,Y,Z)$ and its derivatives along a particle trajectory are calculated by means of a 5-point polynomial interpolation followed by second order off-axis extrapolation (see sections 1.3.2, 1.4.1).

Entrance and/or exit integration boundaries may be defined in the same way as in *CARTEMES* by means of the flag ID and coefficients A, B, C, etc.

CARTEMES : 2-D Cartesian uniform mesh magnetic field map

CARTEMES was originally dedicated to the reading and processing of the measured median plane field maps of the QDD spectrometer SPES2 at Saclay, assuming mid-plane dipole symmetry. However, it can be used for the reading of any 2-D median plane maps, provided that the format of the field data storage file fits the following *FORTRAN* sequence

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
        IF (BINARY) THEN
          READ(NL) (Y(J), J=1, JY)
        ELSE
          READ(NL,FMT='(10F8.2)') (Y(J), J=1, JY)
        ENDIF
        DO 1 I=1, IX
  IF (BINARY) THEN
    READ(NL) X(I), (BMES(I,J), J=1, JY)
  ELSE
    READ(NL,FMT='(10F8.1)') X(I), (BMES(I,J), J=1, JY)
          ENDIF
  1     CONTINUE
```

where, IX and JY are the number of longitudinal and transverse horizontal nodes of the uniform mesh, and $X(I), Y(J)$ their coordinates. *FNAME* is the name of the file containing the field data. For binary files, *FNAME* must begin with 'B_ ' or 'b_', a flag 'BINARY' will thus be set to '.TRUE.' by the *FORTRAN*.

The measured field *BMES* is normalized with *BNORM*,

$$B(I,J) = \mathit{BMES}(I,J) \times \mathit{BNORM}$$

As well the longitudinal coordinate X is normalized with a *XNORM* coefficient (useful to convert to centimeters, the working units in **zgoubi**).
The vector field, $\vec{B}$, and its derivatives out of the median plane are calculated by means of a second or fourth order polynomial interpolation, depending on the value of the parameter *IORDRE* (*IORDRE* = 2, 25 or 4, see section 1.4.2).

In case a particle exits the mesh, its *IEX* flag is set to -1 (see section 5.6.11, p. 184), however it is still tracked with the field being *extrapolated* from the closest nodes of the mesh. Note that such extrapolation process may induce erratic behavior if the distance from the mesh gets too large.

Entrance and/or exit integration boundaries (so-called "droite de coupure" in the code) can be defined with the flag ID, as follows (Fig. 17).

If ID $= 1$: the integration in the field is terminated on a boundary with equation $A'X + B'Y + C' = 0$, and then the trajectories are extrapolated linearly onto the exit border of the map.

If ID $= -1$: an entrance boundary is defined, with equation $A'X + B'Y + C' = 0$, up to which trajectories are first extrapolated linearly from the map entrance border, prior to being integrated in the field.

If ID ≥ 2 : one entrance boundary, and $ID-1$ exit boundaries are defined, as above. The integration in the field terminates on the last ($ID-1$) exit boundary. No extrapolation onto the map exit border is performed in this case.

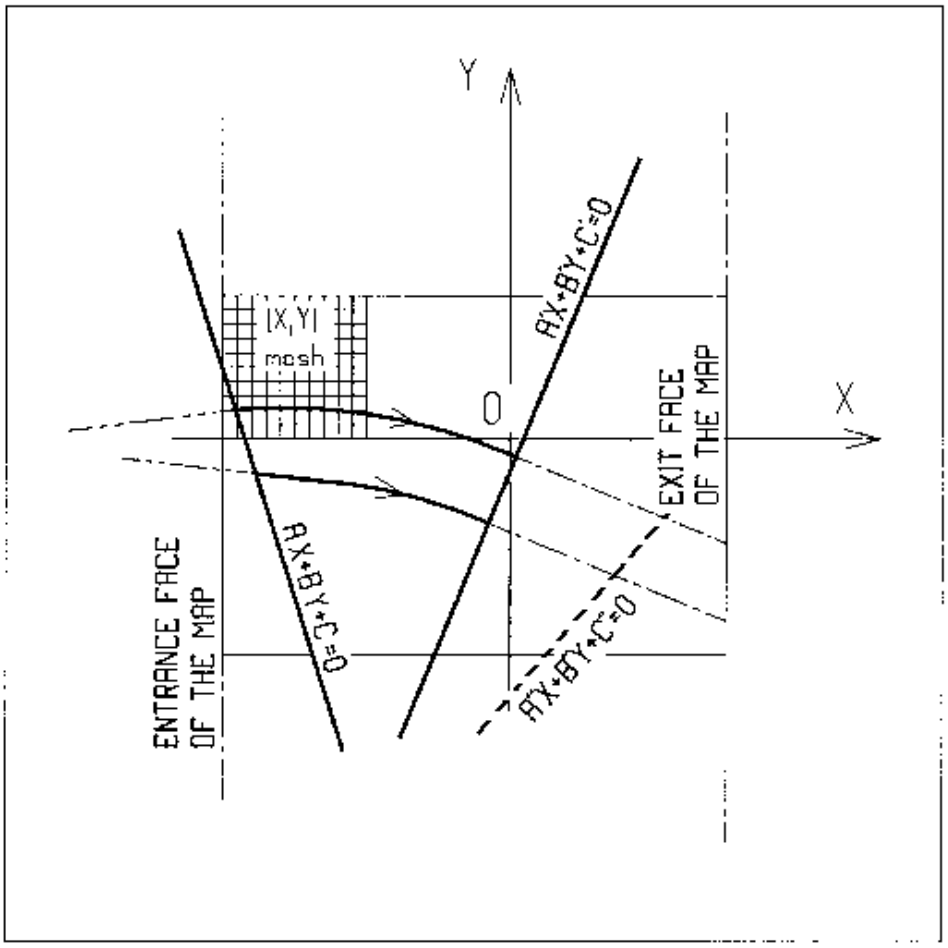


Figure 17: OXY is the coordinate system of the mesh. Integration boundaries may be defined, using $ID \neq 0$: particle coordinates are extrapolated linearly from the entrance face of the map, onto the boundary $A'X+B'Y+C'=0$; after ray-tracing inside the map and terminating on the boundary $AX+BY+C=0$, coordinates are extrapolated linearly onto the exit face of the map if $ID=2$, or terminated on the last ($ID-1$) boundary if $ID>2$.

CAVITE : Accelerating cavity

CAVITE provides a simulation of a (zero length) accelerating cavity ; it can be used in conjunction with keywords *REBELOTE* and *SCALING* for the simulation of multi-turn tracking with synchrotron or fixed field (FFAG, cyclotron) acceleration (see section 5.6.10). It can also be used to simulate recirculating/energy-recovery linacs. *CAVITE* must be preceded by *PARTICUL* for the definition of mass M and charge q.

A major effect of *CAVITE* on optics settings is the following :

The reference rigidity $B\rho_{ref}$ in **zgoubi**, as used for instance when determining optical strengths from field values (see sections 1.2.1-1.2.3), is defined when creating the object by means of *[MC]OBJET*. However, in many cases - options as described below - that reference rigidity will be updated upon crossing a cavity, consistently with the amount of synchronous rigidity increase as induced by the energy change, namely,

$$B\rho_{ref} = \textit{BORO} \longrightarrow B\rho_{ref} = \textit{BORO} + \delta B\rho_s \tag{5.4.8}$$

Note as an illustration of the process, that, in this case, a simple way to have the optical elements have their *strengths* maintained constant is to use *SCALING* with the option *NTIM* $= -1$.

- **If IOPT = 0 :** *CAVITE* is switched off.

- **If IOPT = 1 :** *CAVITE* simulates the RF cavity of a synchrotron accelerator : the periodic motion over $IP = 1$, *NPASS* + 1 turns (passes through the structure) is obtained using the keyword *REBELOTE*, option K = 99, while RF and optical elements time dependent functions are simulated by means of *SCALING* – see section 5.6.10. *CAVITE* may conveniently be located *at the end* of the optical structure, otherwise its phasing has to be indicated. The synchrotron motion of any of the *IMAX* particles of a beam is obtained from the following mapping

$$\begin{cases} \phi_2 - \phi_1 = 2\pi\, f_{RF} \left(\dfrac{\ell}{\beta c} - \dfrac{\mathcal{L}}{\beta_s c} \right) \\ W_2 - W_1 = q\hat{V}\, \sin\phi_1 \end{cases}$$

  where

  ϕ = RF phase ; $\phi_2 - \phi_1$ = variation of ϕ between two traversals
  $\hat{V}$ = peak RF voltage
  W = kinetic energy ; $W_2 - W_1$ = energy gain at a traversal of *CAVITE*
  $\mathcal{L}$ = length of the synchronous closed orbit (to be calculated by prior ray-tracing, see '*CLOSED ORBIT COMPUTATION*' below)
  $\beta_s c$ = velocity associated with the synchronous energy
  ℓ = particle trajectory length between two traversals
  βc = particle velocity
  c = velocity of light.

  The RF frequency f_{RF} is a multiple of the synchronous revolution frequency, and is obtained from the input data, following

$$f_{RF} = \frac{hc}{\mathcal{L}} \frac{q(B\rho)_s}{\sqrt{q^2(B\rho)_s^2 + (Mc)^2}}$$

  where h = harmonic number of the R.F
  $B\rho_s$ = rigidity associated with the synchronous energy

  The synchronous rigidity $(B\rho)_s$ is obtained from the timing law specified by means of *SCALING* following $(B\rho)_s = \textit{BORO} \cdot \textit{SCALE(TIMING)}$ (see *SCALING* for the meaning and calculation of the scale factor *SCALE(TIMING)*). *If SCALING is not used,* $(B\rho)_s$ *is assumed to keep the constant value BORO* as given in the object description (see *OBJET* for instance).

The velocity βc of a particle is calculated from its current rigidity

$$\beta = \frac{q(B\rho)}{\sqrt{q^2(B\rho)^2 + (Mc)^2}}$$

The velocity $\beta_s c$ of the synchronous particle is obtained in the same way from

$$\beta_s = \frac{q(B\rho)_s}{\sqrt{q^2(B\rho)_s^2 + (Mc)^2}}$$

The kinetic energies and rigidities involved in these formulae are related by

$$q(B\rho) = \sqrt{W(W + 2Mc^2)}$$

Finally, the initial conditions for the mapping, at the first turn, are the following

- For the (virtual) synchronous particle

$$\phi_1 = \phi_s = \text{synchronous phase}$$
$$(B\rho)_{1s} = \textit{BORO}$$

- For any of the $I = 1$, *IMAX* particles of the beam

$$\phi_{1I} = \phi_s = \text{synchronous phase}$$
$$(B\rho)_{1I} = \textit{BORO} * D_I$$

where the quantities *BORO* and D_I are given in the object description.

Calculation of the Coordinates

Let $p_I = [p_{XI}^2 + p_{YI}^2 + p_{ZI}^2]^{1/2}$ be the momentum of particle I at the exit of the cavity, while $p_{I_0} = \left[p_{XI_0}^2 + p_{YI_0}^2 + p_{ZI_0}^2\right]^{1/2}$ is its momentum at the entrance. The kick in momentum is assumed to be fully longitudinal, resulting in the following relations between the coordinates at the entrance (denoted by the index zero) and at the exit

$$p_{XI} = \left[p_I^2 - (p_{I_0}^2 - p_{XI_0}^2)\right]^{1/2}$$
$$p_{YI} = p_{YI_0}, \quad \text{and} \quad p_{ZI} = p_{ZI_0} \quad \text{(longitudinal kick)}$$
$$X_I = X_{I_0}, \quad Y_I = Y_{I_0} \quad \text{and} \quad Z_I = Z_{I_0} \quad \text{(zero length cavity)}$$

and for the angles (see Fig. 1)

$$\left.\begin{aligned} T_I &= \text{Atg}\left(\frac{p_{YI}}{p_{XI}}\right) \\ P_I &= \text{Atg}\left(\frac{P_{ZI}}{(p_{XI}^2 + p_{YI}^2)^{1/2}}\right) \end{aligned}\right\} \quad \text{(damping of the transverse motion)}$$

- **If $\mathbf{IOPT = 2}$** : the same simulation of a synchrotron RF cavity as for $\mathbf{IOPT = 1}$ is performed, except that the keyword *SCALING* (family *CAVITE*) is not taken into account in this option : the increase in kinetic energy at each traversal, for the synchronous particle, is

$$\Delta W_s = q\hat{V}\,\sin\phi_s$$

where the synchronous phase ϕ_s is part of the input data to to *CAVITE*. From this, the calculation of the law $(B\rho)_s$ and the RF frequency f_{RF} follows, according to the formulae given in the *IOPT* = 1 case.

IOPT = 2 can be used for SR loss compensation in *storage ring* mode. In that case ϕ_s should be given the appropriate value for compensation for SR induced energy loss.

IOPT = 2 handles double RF systems. Respective harmonic value and voltage amplitudes are h_1, h_2 and V_1, V_2. It follows CERN ISR-TH-RF/80-26 [41]. In particular, the stable phase ϕ_n of the higher harmonic voltage is taken to be

$$\phi_{\rm n} = {\rm atan}(\tan(\phi_{\rm s})/{\rm n})/{\rm n}$$

wherein $n = h_2/h_1$. The synchronous energy gain is

$$\Delta W_s = q\hat{V}\left(\sin(\phi_s) + k\sin(n\phi_n)\right)$$

with $k = V_2/V_1$.

The option *IOPT = 2* handles synchrotron radiation energy loss in the case of stationary bucket (see *IOPT = 11* for accelerating bucket). Compensation is derived from the synchronous phase ϕ_s, via $\Delta W_{SR} = q\hat{V}\ \sin\phi_s$, therefore assuming appropriate value

$$\phi_s = {\rm asin}(\Delta W_{SR}/q\hat{V})$$

with $\Delta W_{SR} =$ the energy loss between passes in the cavity.

- **If IOPT = 3 :** sine RF law, acceleration without synchrotron motion. Any particle will be given a kick

$$\Delta W = q\hat{V}\ \sin\phi_s$$

where $\hat{V}$ and ϕ_s are input data.

The option *IOPT = 3* handles synchrotron radiation energy loss in the case of stationary bucket (as in *IOPT = 2*).

- **If IOPT = 4 :** to be documented.

- **If IOPT = 5 :** to be documented.

- **If IOPT = 6 :** allows reading the RF frequency and/or phase law from an external file (wih name normally "zgoubi.freqLaw.In"). See routines `cavite.f` and `scalin.f` for details Was first used for acceleration in scaling FFAG [49].

- **If IOPT = 7 :** fixed frequency RF, quasi- or isochronous acceleration. Was first used for quasi-isochronous, fixed frequency acceleration in the EMMA prototype linear FFAG [50, 51].

  IOPT = 7 can be used for cyclotron acceleration.

- **If IOPT = 8 :** to be documented.

- **If IOPT = 9 :** to be documented.

- **If IOPT = 10 :** fixed frequency RF. Was first installed/used for CEBAF and eRHIC ERL simulations. A Chambers matrix method is used to get the transverse focusing effect. Particles undergo a longitudinal boost in energy

$$\Delta W = q\hat{V}\ \cos(2\pi f_{RF} t + \phi_s)$$

with t the arrival time at the center of the cavity. The phase datum ϕ_s in this option has two roles, (i) that played in the equation above, (ii) determining the updating of **zgoubi**'s reference rigidity, namely,

$$B\rho_{ref} = \textit{BORO} \rightarrow B\rho_{ref} = \textit{BORO}+\delta B\rho_s \quad \text{with} \quad \delta B\rho_s = \frac{\Delta(\beta_s(W_s+M))}{qc}, \quad \Delta W_s = q\hat{V}\ \cos(\phi_s) \tag{5.4.9}$$

The cavity length L as specified in the input data is also used to update the path length of the particles (linearly). An option ('IOP' flag) allows small ΔW approximation of Chambers matrices, as well as removing the longitudinal damping (thus geometrical emittance is conserved), as follows :

- IOP = 1 : Chambers cavity with $\Delta W/W$ aproximation, namely, for both planes,

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{out} = \begin{pmatrix} \sqrt{(W_i/W_o)} & L_{cav} \times \sqrt{(W_i/W_o)} \\ 0. & \sqrt{(W_i/W_o)} \end{pmatrix} \begin{pmatrix} x \\ x' \end{pmatrix}_{in} \tag{5.4.10}$$

  with W_i, W_o respectively the incoming and outgoing kinetic energies.

- IOP = -1 : Same as 1, with matrix in both planes re-normalized to determinant=1 (*i.e.*, $R_{ij} \rightarrow R_{ij}/\sqrt{R_{11}R_{12} - R_{21}R_{22}}$),

- IOP = 2 : Chambers cavity without aproximation on $\Delta W/W$, namely, for both planes,

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{out} = \begin{pmatrix} \cos(u) - \sqrt{2}\sin(u)\cos(\phi) & vW_i \sin(u)\cos(\phi) \\ -\frac{\sin(u)}{vW_o}(2\cos(\phi) + \frac{1}{\cos(\phi)}) & \frac{W_i}{W_o}(\cos(u) + \sqrt{2}\sin(u)\cos(\phi)) \end{pmatrix} \begin{pmatrix} x \\ x' \end{pmatrix}_{in} \tag{5.4.11}$$

  with $u = \log(W_o/W_i)/(\sqrt{8}\cos(\phi))$, $v = \sqrt{8}L/(W_o - W_i)$, ϕ the particle phase at the cavity.

- IOP = -2 : Same as 1, with matrix in both planes re-normalized to determinant=1.

- IOP = 0 : Transverse matrix is that of a drift with length L.

- **If $\mathbf{IOPT = 11}$:** under development.

The same simulation of a synchrotron RF cavity as for $\mathbf{IOPT = 2}$ is performed, but in addition *IOPT = 10* handles synchrotron radiation energy loss, both stationnary or accelerated bucket.

CLOSED ORBIT COMPUTATION

Due to possible dipole type of optical defects (*e.g.*, fringe fields, straight axis combined function dipoles), the closed orbit may not coincide with the ideal axis of the optical elements (hence it will be almost everywhere non-zero). One way to calculate it at the beginning of the structure (*i.e.*, where the initial particle coordinates are defined) is to ray-trace a single particle over a sufficiently large number of turns, starting with initial conditions taken near the reference orbit, so as to obtain statistically well-defined transverse phase-space ellipses. The local closed orbit coincides with the coordinates Y_c, T_c, Z_c, P_c of the center of the ellipses. A few iterations are usually sufficient (avoid near-integer tunes) to ensure accuracy. Next, ray-tracing over one turn a particle starting with the initial condition (Y_c, T_c, Z_c, P_c) will provide the entire closed orbit, and as a sub-product its length $\mathcal{L}$ (the $F(6,1)$ coordinate in the *FORTRAN*).

RECOVERING FROM A CRASH

When accelerating a bunch in a pulsed ring, *SCALING* is used to ramp magnetic fields, so following the rigidity increase by *CAVITE*. *CAVITE* may or may not be part of the *SCALING* argument list, this is discussed in the *SCALING* section, page 82. Now, if the job is stopped - due to computer crash or else - it is possible to resume the tracking from the latest records (in [b_]zboubi.fai storage file), for instance using *OBJET*, *KOBJ*= 3. In the presence of acceleration using *CAVITE* however, since there is no reference particle in **zgoubi**, *CAVITE* needs be told the reference rigidity at the corresponding time/turn where tracking is resumed. This is done using *SCALING*. This has the effect of updating the refrence rigidity $B\rho_{ref}$ to its correct value (see Note 1 above), the one it had when the job stopped. See the *SCALING* section for more (page 80) and the examples there, as well as the example page 255.

PRINT : The optional command 'PRINT', following the first argument, *'IOPT'*, in *CAVITE* keyword, will cause logging of computational data (synchronous time and momentum, RF phase, and much more) in the file zgoubi.CAVITE.Out.

For software developers

Cavity simulations are done in cavite.f. Management of the reference rigidity $B\rho_{ref}$ is ensured for part in *cavite.f*, and for part in routines associated with '*SCALLING'*, essentially *scaler.f.*

CHAMBR : Long transverse aperture limitation

CHAMBR causes the identification, counting and stopping of particles that reach the transverse limits of the vacuum chamber. The chamber can be either rectangular (*IFORM* = 1) or elliptic (*IFORM* = 2). The chamber is centered at *YC*, *ZC* and has transverse dimensions $\pm YL$ and $\pm ZL$ such that any particle will be stopped if its coordinates Y, Z satisfy

$$(Y - \mathit{YC})^2 \geq YL^2 \text{ or } (Z - \mathit{ZC})^2 \geq ZL^2 \quad \text{if} \quad \mathit{IFORM} = 1$$
$$\frac{(Y - \mathit{YC})^2}{YL^2} + \frac{(Z - \mathit{ZC})^2}{ZL^2} \geq 1 \quad \text{if} \quad \mathit{IFORM} = 2$$

The conditions introduced with *CHAMBR* are valid along the optical structure until the next occurrence of the keyword *CHAMBR*. Then, if $IL = 1$ the aperture is possibly modified by introducing new values of *YC*, *ZC*, YL and ZL, or, if $IL = 2$ the chamber ends and information is printed concerning those particles that have been stopped.

The testing is done in optical elements at each integration step, between the *EFB*s. For instance, in *QUADRUPO* there will be no testing from $-X_E$ to 0 and from *XL* to $\mathit{XL} + X_S$, but only from 0 to *XL* ; in *DIPOLE*, there is no testing as long as the *ENTRANCE EFB* is not reached, and testing is stopped as soon as the *EXIT* or *LATERAL EFB*s are passed.

In optical elements defined in polar coordinates, Y stands for the radial coordinate (*e.g.*, *DIPOLE*, see Figs. 3C, p. 29, and 11, p. 92). Thus, centering *CHAMBR* at
$YC = \mathit{RM}$ simulates a chamber curved with radius *RM*, and having a radial acceptance $\mathit{RM} \pm \mathit{YL}$. In *DRIFT*, the testing is done at the beginning and at the end, and only for positive drifts. There is no testing in *CHANGREF*.

When a particle is stopped, its index *IEX* (see *OBJET* and section 5.6.11) is set to the value -4, and its actual path length is stored in the array *SORT* for possible further use.

CHANGREF : Transformation to a new reference frame

CHANGREF transports particles from a reference plane (O, Y, Z) at path distance S, to a new one by a combination of translations and/or rotations. It essentially aims at positioning optical elements with respect to one another, as setting a reference frame at the entrance or exit of field maps, or to simulate misalignments (see also *KPOS* option). *CHANGREF* can be placed anywhere in a structure.

Spin tracking, particle decay and gas-scattering are taken into account in *CHANGREF*. Energy loss by synchrotron radiation (*SRLOSS* keyword) is not.

There are two "styles" of *CHANGREF*, as follows.

The "old style" *CHANGREF* requires the three data *XCE*, *YCE*, *ALE* and then gets the new particle coordinates Y_2, T_2, Z_2, P_2 and path length S_2 from the old ones Y_1, T_1, Z_1, P_1 and S_1 using

$$
\begin{aligned}
T_2 &= T_1 - ALE \\
Y_2 &= \frac{(Y_1 - YCE)\cos T_1 + XCE \sin T_1}{\cos T_2} \\
DL^2 &= (XCE - Y_2 \sin ALE)^2 + (YCE - Y_1 + Y_2 \cos ALE)^2 \\
Z_2 &= Z_1 + DL \mathrm{tg} P_1 \\
S_2 &= S_1 + \frac{DL}{\cos P_1} \\
P_2 &= P_1
\end{aligned}
$$

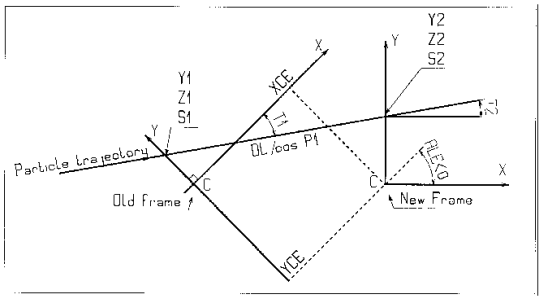


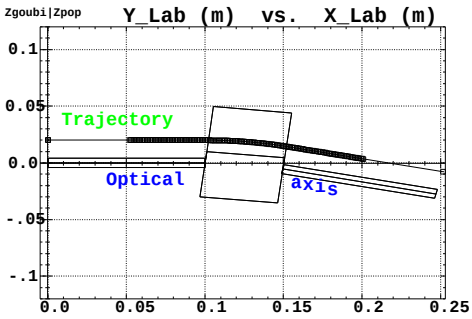
Figure 18: Scheme of the *CHANGREF* procedure.

where, XCE and YCE are shifts in the horizontal plane along, respectively, X- and Y-axis, and ALE is a rotation around the Z-axis. DL is given the sign of $XCE - Y_2 \sin(ALE)$.

The example below shows the use of *CHANGREF* for the symmetric positioning of a combined function dipole+quadrupole magnet in a drift-bend-drift geometry with 12.691 degrees deviation (obtained upon combined effect of a dipole component and of quadrupole axis shifted 1 cm off optical axis).

Zgoubi data file :

```
Using CHANGREF, "Old style"
'OBJET'
51.71103865921708          ! Electron, Ekin=15MeV.
2
1 1                        ! One particle, with
2. 0.  0.0 0.0 0.0 1. 'R' ! Y_0=2 cm.
1 1 1 1 1 1 1
'MARKER'    BEG    .plt    ! Print into zgoubi.plt.
'DRIFT'                    ! 10 cm drift.
10.
'CHANGREF'
0. 0. -6.34165             ! 1/ half Z-rotation.
'CHANGREF'
0. 1. 0.                   ! 2/ Y-shift.
'MULTIPOL'       ! Combined function dipole + quadrupole.
   2                       ! Print into zgoubi.plt.
5  10. 2.064995867082342  2. 0. 0. 0. 0. 0. 0. 0.
 0 0  5. 1.1  1.00 1.00 1.00 1.00 1.00 1. 1. 1. 1.
4  .1455   2.2670  -.6395  1.1558  0. 0.
 0 0  5. 1.1  1.00 1.00 1.00 1.00 1.00 1. 1. 1. 1.
4  .1455   2.2670  -.6395  1.1558  0. 0.
0 0 0 0 0 0 0 0 0 0
.1   step size
1  0. 0.  0.
'CHANGREF'
0. -1. -6.34165            ! 1/ Y-shift, 2/ half Z-rotate.
'DRIFT'                    ! 10 cm drift.
10.
'FAISCEAU'
'END'
```



Note : The square markers scheme the stepwise integration in case of ± 5 cm additional fringe field extent upstream and downstream of the 5 cm long multipole.

The “new style” *CHANGREF* allows all 6 degrees of freedom rather than just 3, namely, X-, Y-, Z-shift, X-, Y-, Z-rotation. In addition, *CHANGREF* “new style” allows up to 9 successive such elementary transformations, in arbitrary order. The “old style” example above is transposed into “new style”, hereafter.

Zgoubi data file :

```
Using CHANGREF, "New Style"
'OBJET'
51.71103865921708         ! Electron, Ekin=15MeV.
2
1 1                       ! One particle, with
2. 0.  0.0 0.0 0.0 1. 'R' ! Y_0=2 cm.
1 1 1 1 1 1
'MARKER'    BEG    .plt   ! Print into zgoubi.plt.
'DRIFT'                   ! 10 cm drift.
10.
'CHANGREF'
ZR -6.34165 YS 1.         ! 1/ half Z-rotate, 2/ Y-shift.
'MULTIPOL'       ! Combined function dipole + quadrupole.
   2                      ! Print into zgoubi.plt.
5  10. 2.064995867082342  2. 0. 0. 0. 0. 0. 0. 0.
 0 0  5. 1.1  1.00 1.00 1.00 1.00 1.00 1. 1. 1. 1.
4  .1455   2.2670  -.6395  1.1558  0. 0.
 0 0  5. 1.1  1.00 1.00 1.00 1.00 1.00 1. 1. 1. 1.
4  .1455   2.2670  -.6395  1.1558  0. 0.
0 0 0 0 0 0 0 0 0 0
.1   step size
1  0. 0.  0.
'CHANGREF'
YS -1. ZR -6.34165        ! 1/ Y-shift, 2/ half Z-rotate.
'DRIFT'                   ! 10 cm drift.
10.
'FAISCEAU'
'END'
```

CIBLE or TARGET : Generate a secondary beam following target interaction

The reaction is $1 + 2 \longrightarrow 3 + 4$ with the following parameters

Laboratory momentum	$p_1 \equiv 0$	p_2	p_3	p_4
Rest mass	M_1	M_2	M_3	M_4
Total energy in laboratory	M_1c^2	W_2	W_3	W_4

The geometry of the interaction is shown in Fig. 19.

The angular sampling at the exit of the target consists of the NT coordinates 0, $\pm TS$, $\pm 2 * TS$... $\pm(NT - 1) * TS/2$ in the median plane, and the NP coordinates 0, $\pm PS$, $\pm 2 * PS$... $\pm(NP - 1) * PS/2$ in the vertical plane.

The position of B downstream is deduced from that of A upstream by a transformation equivalent to two transformations using *CHANGREF,* namely

$$\textit{CHANGREF}(XCE = YCE = 0, \quad \textit{ALE} = \beta)$$

followed by

$$\textit{CHANGREF}(XCE = YCE = 0, \quad \textit{ALE} = \theta - \beta).$$

Particle 4 is discarded, while particle 3 continues. The energy loss Q is related to the variable mass M_4 by

$$Q = M_1 + M_2 - (M_3 + M_4) \qquad \text{and} \qquad dQ = -dM_4$$

The momentum sampling of particle 3 is derived from conservation of energy and momentum, according to

$$\begin{aligned} M_1c^2 + W_2 &= W_3 + W_4 \\ p_4^2 &= p_2^2 + p_3^2 - 2p_2p_3\cos(\theta - T) \end{aligned}$$

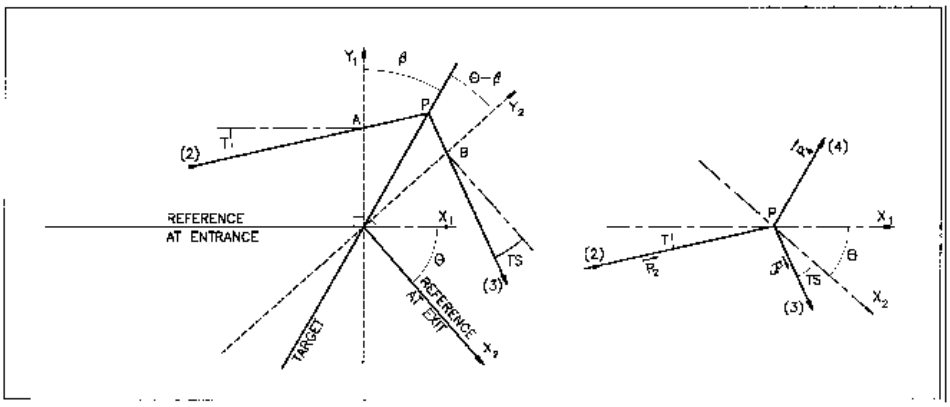


Figure 19: Scheme of the principles of *CIBLE (TARGET)*
A, T = position, angle of incoming particle 2 in the entrance reference frame
P = position of the interaction
B, T = position, angle of the secondary particle in the exit reference frame
θ = angle between entrance and exit frames
β = tilt angle of the target

COLLIMA : Collimator

COLLIMA acts as a mathematical aperture of zero length. It causes the identification, counting and stopping of particles that reach the aperture limits.

Physical Aperture

A physical aperture can be either rectangular (*IFORM* = 1) or elliptic (*IFORM* = 2). The collimator is centered at *YC, ZC* and has transverse dimensions $\pm YL$ and $\pm ZL$ such that any particle will be stopped if its coordinates Y, Z satisfy

$$(Y - YC)^2 \geq YL^2 \text{ or } (Z - ZC)^2 \geq ZL^2 \quad \text{if} \quad IFORM = 1$$
$$\frac{(Y - YC)^2}{YL^2} + \frac{(Z - ZC)^2}{ZL^2} \geq 1 \quad \text{if} \quad IFORM = 2$$

Longitudinal Phase-space Collimation

COLLIMA can act as a longitudinal phase-space aperture, coordinates acted on are selected with *IFORM.J.* Any particle will be stopped if its horizontal (h) and vertical (v) coordinates satisfy

$$(h \leq h_{min} \text{ or } h \geq h_{max}) \text{ or } (v \leq v_{min} \text{ or } v \geq v_{max})$$

wherein, h is either path length S if *IFORM*=6 or time if *IFORM*=7, and v is either 1+DP/P if *J*=1 or kinetic energy if *J*=2 (provided mass and charge have been defined using the keyword *PARTICUL*).

Transverse Phase-space Collimation

COLLIMA can act as a transverse phase-space aperture. Any particle will be stopped if its coordinates satisfy

$$\gamma_Y Y^2 + 2\alpha_Y YT + \beta_Y T^2 \geq \epsilon_Y/\pi \quad \text{if} \quad IFORM = 11 \text{ or } 14$$
$$\gamma_Z Z^2 + 2\alpha_Z ZP + \beta_Z P^2 \geq \epsilon_Z/\pi \quad \text{if} \quad IFORM = 12 \text{ or } 15$$

If *IFORM*=11 (respectively 12) then ϵ_Y/π (respectively ϵ_Z/π) is to be specified by the user as well as $\alpha_{Y,Z}$, $\beta_{Y,Z}$. If *IFORM*=14 (respectively 15) then α_Y and β_Y (respectively α_Z, β_Z) are determined by **zgoubi** by prior computation of the matched ellipse to the particle population, so only $\epsilon_{Y,Z}/\pi$ need be specified by the user.

When a particle is stopped, its index *IEX* (see *OBJET* and section 5.6.11) is set to the value -4, and its actual path length is stored in the array *SORT* for possible further use with *HISTO*).

DECAPOLE : Decapole magnet (Fig. 20)

The meaning of parameters for *DECAPOLE* is the same as for *QUADRUPO.*
In fringe field regions the magnetic field $\vec{B}(X,Y,Z)$ and its derivatives up to fourth order are derived from the scalar potential expressed to the 5th order in Y and Z

$$V(X,Y,Z) = G(X)\left(Y^4Z - 2Y^2Z^3 + \frac{Z^5}{5}\right)$$
$$\text{with } G_0 = \frac{B_0}{R_0^4}$$

The modelling of the fringe field form factor $G(X)$ is described under *QUADRUPO*, p. 147.

Outside fringe field regions, or everywhere in sharp edge decapole ($\lambda_E = \lambda_S = 0$) , $\vec{B}(X,Y,Z)$ in the magnet is given by

$$B_X = 0$$
$$B_Y = 4G_0(Y^2 - Z^2)YZ$$
$$B_Z = G_0(Y^4 - 6Y^2Z^2 + Z^4)$$

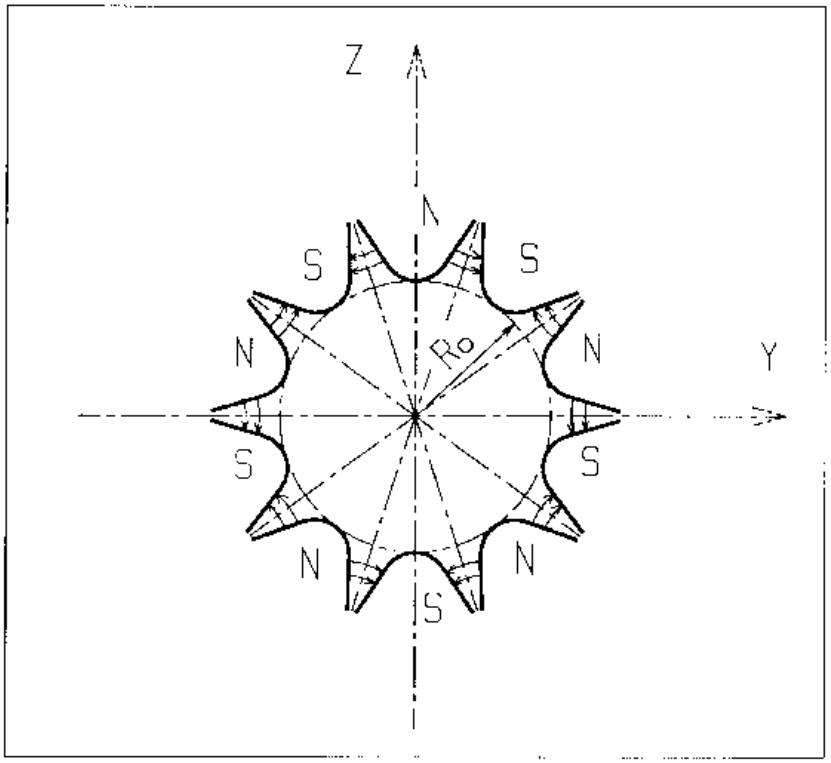


Figure 20: Decapole magnet

DIPOLE : Dipole magnet, polar frame

DIPOLE provides a model of a dipole field, and allows transverse indices. The field along a particle trajectory is computed as the particle motion proceeds, straightforwardly from the dipole geometrical boundaries. Field simulation in *DIPOLE* is the same as used in *DIPOLE-M* and *AIMANT* for computing a field map ; the essential difference in *DIPOLE* is in its skipping that intermediate stage of field map generation found in *DIPOLE-M* and *AIMANT*.

DIPOLE is available in a version, *DIPOLES*, that allows overlapping of fringe fields in a configuration of neighboring magnets.

The dimensioning of the magnet is defined by (Fig. 11, p. 92)

AT : total angular aperture
RM : mean radius used for the positioning of field boundaries

The 2 or 3 effective field boundaries (EFB), from which the dipole field is drawn, are defined from geometric boundaries, the shape and position of which are determined by the following parameters.

ACENT : arbitrary inner angle, used for EFBs positioning
ω : azimuth of an EFB with respect to *ACENT*
θ : angle of an EFB with respect to its azimuth (wedge angle)
R_1, R_2 : radius of curvature of an EFB
U_1, U_2 : extent of the linear part of an EFB.

The magnetic field is calculated in polar coordinates. At any position (R, θ) along the particle trajectory the value of the vertical component of the mid-plane field is calculated using

$$B_Z(R,\theta) = \mathcal{F}(R,\theta) * B_0 * \left(1 + N * \left(\frac{R-RM}{RM}\right) + B * \left(\frac{R-RM}{RM}\right)^2 + G * \left(\frac{R-RM}{RM}\right)^3\right) \tag{5.4.12}$$

where N, B and G are respectively the first, second and third order field indices and $\mathcal{F}(R,\theta)$ is the fringe field coefficient (it determines the "flutter" in periodic structures).

Calculation of the Fringe Field Coefficient

With each EFB a realistic extent of the fringe field, λ (normally equal to the gap size), is associated and a fringe field coefficient F is calculated. In the following λ stands for either λ_E (Entrance), λ_S (Exit) or λ_L (Lateral EFB).

F is an exponential type fringe field (Fig. 12, p. 94) given by [37]

$$F = \frac{1}{1 + \exp P(s)}$$

wherein s is the distance to the EFB and depends on (R, θ), and

$$P(s) = C_0 + C_1\left(\frac{s}{\lambda}\right) + C_2\left(\frac{s}{\lambda}\right)^2 + C_3\left(\frac{s}{\lambda}\right)^3 + C_4\left(\frac{s}{\lambda}\right)^4 + C_5\left(\frac{s}{\lambda}\right)^5$$

It is also possible to simulate a shift of the *EFB*, by giving a non zero value to the parameter *shift*. s is then changed to $s-$*shift* in the previous equation. This allows small variations of the magnetic length.

Let F_E (respectively F_S, F_L) be the fringe field coefficient attached to the entrance (respectively exit, lateral) EFB. At any position on a trajectory the resulting value of the fringe field coefficient (eq. 5.4.12) is

$$\mathcal{F}(R,\theta) = F_E * F_S * F_L$$

In particular, $F_L \equiv 1$ if no lateral EFB is requested.

Calculation of the Mid-plane Field and Derivatives

$B_Z(R,\theta)$ in Eq. 5.4.12 is computed at the $n \times n$ nodes ($n = 3$ or 5 in practice) of a "flying" interpolation grid in the median plane centered on the projection m_0 of the actual particle position M_0 as schemed in Fig. 21. A polynomial interpolation is involved, of the form

$$B_Z(R,\theta) = A_{00} + A_{10}\theta + A_{01}R + A_{20}\theta^2 + A_{11}\theta R + A_{02}R^2$$

that yields the requested derivatives, using

$$A_{kl} = \frac{1}{k!l!}\frac{\partial^{k+l}B_Z}{\partial\theta^k \partial r^l}$$

Note that, the source code contains the explicit analytical expressions of the coefficients A_{kl} solutions of the normal equations, so that the operation *is not* CPU time consuming.

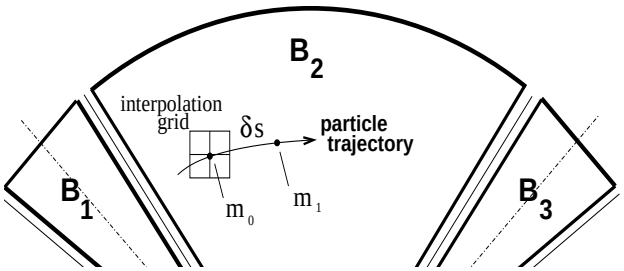


Figure 21: Interpolation method. m_0 and m_1 are the projections in the median plane of particle positions M_0 and M_1 and separated by δs, projection of the integration step.

Extrapolation Off Median Plane

From the vertical field $\vec{B}$ and derivatives in the median plane, first a transformation from polar to Cartesian coordinates is performed, following eqs (1.4.9 or 1.4.10), then, extrapolation off median plane is performed by means of Taylor expansions, following the procedure described in section 1.3.3.

DIPOLE-M : Generation of dipole mid-plane 2-D map, polar frame

DIPOLE-M is a more recent, simpler and improved version of *AIMANT*.

The keyword *DIPOLE-M* provides an automatic generation of a dipole field map in polar coordinates. The extent of the map is defined by the following parameters, as shown in Figs. 11A and 11B.

AT	:	total angular aperture
RM	:	mean radius used for the positioning of field boundaries
RMIN, RMAX	:	minimum and maximum radii

The 2 or 3 effective field boundaries (EFB) inside the map are defined from geometric boundaries, the shape and position of which are determined by the following parameters.

ACENT	:	arbitrary inner angle, used for EFBs positioning
ω	:	azimuth of an EFB with respect to *ACENT*
θ	:	angle of an EFB with respect to its azimuth (wedge angle)
R_1, R_2	:	radius of curvature of an EFB
U_1, U_2	:	extent of the linear part of an EFB.

At any node of the map mesh, the value of the field is calculated as

$$B_Z(R,\theta) = \mathcal{F}(R,\theta) * B_0 * \left(1 + N * \left(\frac{R-RM}{RM}\right) + B * \left(\frac{R-RM}{RM}\right)^2 + G * \left(\frac{R-RM}{RM}\right)^3\right) \tag{5.4.13}$$

where N, B and G are respectively the first, second and third order field indices and $\mathcal{F}$ is the fringe field coefficient.

Calculation of the Fringe Field Coefficient

With each EFB a realistic extent of the fringe field, λ (normally equal to the gap size), is associated and a fringe field coefficient F is calculated. In the following λ stands for either λ_E (Entrance), λ_S (Exit) or λ_L (Lateral EFB).

F is an exponential type fringe field (Fig. 12) given by [37]

$$F = \frac{1}{1 + \exp P(s)}$$

where s is the distance to the EFB, and

$$P(s) = C_0 + C_1\left(\frac{s}{\lambda}\right) + C_2\left(\frac{s}{\lambda}\right)^2 + C_3\left(\frac{s}{\lambda}\right)^3 + C_4\left(\frac{s}{\lambda}\right)^4 + C_5\left(\frac{s}{\lambda}\right)^5$$

It is also possible to simulate a shift of the *EFB*, by giving a non zero value to the parameter *shift*. s is then changed to $s - $ *shift* in the previous equation. This allows small variations of the total magnetic length.

Let F_E (respectively F_S, F_L) be the fringe field coefficient attached to the entrance (respectively exit, lateral) EFB. At any node of the map mesh, the resulting value of the fringe field coefficient (eq. 5.4.13) is

$$\mathcal{F}(R,\theta) = F_E * F_S * F_L$$

In particular, $F_L \equiv 1$ if no lateral EFB is requested.

The Mesh of the Field Map

The magnetic field is calculated at the nodes of a mesh with polar coordinates, in the median plane. The radial step is given by

$$\delta R = \frac{RMAX - RMIN}{IRMAX - 1}$$

and the angular step by

$$\delta\theta = \frac{AT}{IAMAX - 1}$$

where *RMIN*and *RMAX* are the lower and upper radial limits of the field map, and AT is its total angular aperture (Fig. 11B). *IRMAX* and *IAMAX* are the total number of nodes in the radial and angular directions.

Simulating Field Defects and Shims

Once the initial map is calculated, it is possible to modify it by means of the parameter *NBS*, so as to simulate field defects or shims.

If $\mathbf{NBS = -2}$, the map is globally modified by a perturbation proportional to $R - R_0$, where R_0 is an arbitrary radius, with an amplitude $\Delta B_Z/B_0$, so that B_Z at the nodes of the mesh is replaced by

$$B_Z * \left(1 + \frac{\Delta B_Z}{B_0}\,\frac{R - R_0}{RMAX\text{- }RMIN}\right)$$

If $\mathbf{NBS = -1}$, the perturbation is proportional to $\theta - \theta_0$, and B_Z is replaced by

$$B_Z * \left(1 + \frac{\Delta B_Z}{B_0}\,\frac{\theta - \theta_0}{AT}\right)$$

If $\mathbf{NBS \geq 1}$, then *NBS* shims are introduced at positions $\frac{R_1 + R_2}{2}$, $\frac{\theta_1 + \theta_2}{2}$ (Fig. 14) [38]
The initial field map is modified by shims with second order profiles given by

$$\theta = \left(\gamma + \frac{\alpha}{\mu}\right)\,\beta\,\frac{X^2}{\rho^2}$$

where X is shown in Fig. 12, $\rho = \frac{R_1 + R_2}{2}$ is the central radius, α and γ are the angular limits of the shim, β and μ are parameters.
At each shim, the value of B_Z at any node of the initial map is replaced by

$$B_Z * \left(1 + F\theta * FR * \frac{\Delta B_Z}{B_0}\right)$$

where $F\theta = 0$ or $FR = 0$ outside the shim, and $F\theta = 1$ and $FR = 1$ inside.

Extrapolation Off Median Plane

The vector field $\vec{B}$ and its derivatives in the median plane are calculated by means of a second or fourth order polynomial interpolation, depending on the value of the parameter *IORDRE* (*IORDRE*=2, 25 or 4, see section 1.4.2). The transformation from polar to Cartesian coordinates is performed following eqs (1.4.9 or 1.4.10). Extrapolation off median plane is then performed by means of Taylor expansions, following the procedure described in section 1.3.3.

DIPOLES : Dipole magnet N-tuple, polar frame [42, 43]

DIPOLES works much like *DIPOLE* as to the field modelling, yet with the particularity that it allows positioning up to 5 such dipoles within the angular sector with full aperture AT thus allowing accounting for overlapping fringe fields. This is done in the following way[5].

The dimensioning of the magnet is defined by

AT : total angular aperture
RM : mean radius used for the positioning of field boundaries

For each one of the $N = 1$ to 5 dipoles of the N-tuple, the 2 effective field boundaries (entrance and exit EFBs) from which the dipole field (eqs. 5.4.14, 5.4.15) is drawn are defined from geometrical boundaries, the shape and position of which are determined by the following parameters (in the same manner as in *DIPOLE, DIPOLE-M*) (see Fig. 11-A, p. 92, and Fig. 22)

ACN_i : arbitrary inner angle, used for EFBs positioning
ω : azimuth of an EFB with respect to *ACN*
θ : angle of an EFB with respect to its azimuth (wedge angle)
R_1, R_2 : radius of curvature of an EFB
U_1, U_2 : extent of the linear part of an EFB

Calculation of the Field From a Single Dipole

The magnetic field is calculated in polar coordinates. At all (R, θ) in the median plane ($Z = 0$), the magnetic field due a single one (index i) of the dipoles of a N-tuple magnet can take either form, upon option,

$$(i) \quad B_{Zi}(R,\theta) = B_{Z0,i}\,\mathcal{F}_i(R,\theta)\,(1 + b_{1_i}(R - RM_i)/RM_i + b_{2_i}(R - RM_i)^2/RM_i^2 + ...) \quad (5.4.14)$$

$$(ii) \quad B_Z(R,\theta) = B_{Z0,i} + \textstyle\sum_{i=1}^{N} \mathcal{F}_i(R,\theta)\,(b_{1_i}(R - RM_i) + b_{2_i}(R - RM_i)^2 + ...) \quad (5.4.15)$$

wherein $B_{Z0,i}$ is a reference field, at reference radius RM_i, and $\mathcal{F}(R,\theta)$ is the fringe field coefficient, see below. This field model is proper to simulate for instance chicane dipoles, cyclotron or FFAG magnets, etc.

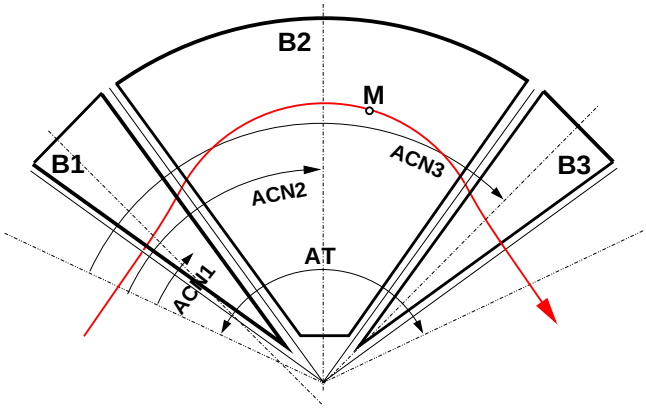


Figure 22: Definition of a dipole triplet using the *DIPOLES* or *FFAG* procedures.

Calculation of the Fringe Field Coefficient

In a dipole, a realistic extent of the fringe field, g, is associated with each EFB, and a fringe field coefficient F is calculated.

[5] *FFAG* can be referred to as another instance of a procedure based on such method.

F is an exponential type fringe field (Fig. 12, page 94) given by [37]

$$F = \frac{1}{1 + \exp P(d)}$$

wherein d is the distance to the EFB and depends on (R, θ), and

$$P(d) = C_0 + C_1 \left(\frac{d}{g}\right) + C_2 \left(\frac{d}{g}\right)^2 + C_3 \left(\frac{d}{g}\right)^3 + C_4 \left(\frac{d}{g}\right)^4 + C_5 \left(\frac{d}{g}\right)^5$$

In addition, g is made dependent of R (a way to simulate the effect of variable gap size on fringe field extent), under the form

$$g(R) = g_0 \left(RM/R\right)^{\kappa}$$

This dependence is accounted for rigorously if the interpolation method (see below) is used, or else to order zero (derivatives of $g(R)$ are not considered) if the analytic method (below) is used.

Let F_E (respectively F_S) be the fringe field coefficient attached to the entrance (respectively exit) EFB ; at any position on a trajectory the resulting value of the fringe field coefficient is taken to be

$$\mathcal{F}_i(R, \theta) = F_E * F_S \tag{5.4.16}$$

Calculation of the Field Resulting From all N Dipoles

Now, accounting for N neighboring dipoles in an N-tuple, the mid-plane field and field derivatives are obtained by addition of the contributions of the N dipoles taken separately, namely

$$B_Z(R, \theta) = \sum_{i=1,N} B_{Zi}(R, \theta) \tag{5.4.17}$$

$$\frac{\partial^{k+l} \vec{B_Z}(R, \theta)}{\partial \theta^k \partial r^l} = \sum_{i=1,N} \frac{\partial^{k+l} \vec{B_{Zi}}(R, \theta)}{\partial \theta^k \partial r^l} \tag{5.4.18}$$

Note that, in doing so it is not meant that field superposition does apply in reality, it is just meant to provide the possibility of obtaining a realistic field shape, that would for instance closely match (using appropriate $C_0 - C_5$ sets of coefficients) 3-D field simulations obtained from magnet design codes.

Calculation of the Mid-plane Field Derivatives

Two methods have been implemented to calculate the field derivatives in the median plane (Eq. 5.4.17), based on either analytical expressions derived from the magnet geometrical description, or classical numerical interpolation.

The first method has the merit of insuring best symplecticity in principle and fastest tracking. The interest of the second method is in its facilitating possible changes in the mid-plane magnetic field model $B_Z(R, \theta)$, for instance if simulations of shims, defects, or special R, θ field dependence need to be introduced.

Analytical method [44] :

The starting ingredients are, on the one hand distances to the EFBs,

$$d(R, \theta) = \sqrt{(x(R, \theta) - x_0(R, \theta))^2 + (y(R, \theta) - y_0(R, \theta))^2}$$

to be computed for the two cases d_{Entrance}, d_{Exit}, and on the other hand the expressions of the coordinates of particle position M and its projection P on the EFB in terms of the magnet geometrical parameters, namely

$$\begin{aligned} x(R,\theta) &= \cos(ACN-\theta) - RM \\ y(R,\theta) &= R\sin(ACN-\theta) \\ x_P(R,\theta) &= \sin(u)\,(y(R,\theta)-y_b)/2 + x_b\,\sin^2(u) + x(R,\theta)\,\cos^2(u) \\ y_P(R,\theta) &= \sin(u)\,(x(R,\theta)-x_b)/2 + y_b\,\cos^2(u) + y(R,\theta)\,\sin^2(u) \end{aligned}$$

with $x_b,\ y_b,\ u$ parameters drawn from the magnet geometry (sector angle, wedge angle, face curvatures, etc.).

These ingredients allow calculating the derivatives $\frac{\partial^{u+v}x(R,\theta)}{\partial\theta^u\partial r^v}, \frac{\partial^{u+v}y(R,\theta)}{\partial\theta^u\partial r^v}, \frac{\partial^{u+v}x_0(R,\theta)}{\partial\theta^u\partial r^v}, \frac{\partial^{u+v}y_0(R,\theta)}{\partial\theta^u\partial r^v}$, which, in turn, intervene in the derivatives of the compound functions $\frac{\partial^{u+v}F(R,\theta)}{\partial\theta^u\partial r^v}$, $\frac{\partial^{u+v}p(R,\theta)}{\partial\theta^u\partial r^v}$, $\frac{\partial^{u+v}d(R,\theta)}{\partial\theta^u\partial r^v}$.

Interpolation method :

The expression $B_Z(R,\theta)$ in Eq. 5.4.17 is, in this case, computed at the $n\times n$ nodes ($n=3$ or 5 in practice) of a "flying" interpolation grid in the median plane centered on the projection m_0 of the actual particle position M_0 as schemed in Fig. 23. A polynomial interpolation is involved, of the form

$$B_Z(R,\theta) = A_{00} + A_{10}\theta + A_{01}R + A_{20}\theta^2 + A_{11}\theta R + A_{02}R^2$$

that yields the requested derivatives, using

$$A_{kl} = \frac{1}{k!l!}\,\frac{\partial^{k+l}B}{\partial\theta^k\partial r^l}$$

Note that, the source code contains the explicit analytical expressions of the coefficients A_{kl} solutions of the normal equations, so that the operation *is not* CPU time consuming.

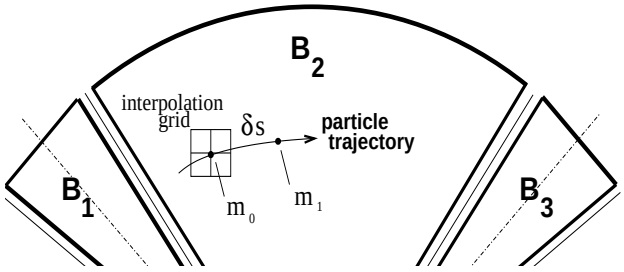


Figure 23: Interpolation method. m_0 and m_1 are the projections in the median plane of particle positions M_0 and M_1 and separated by δs, projection of the integration step.

Extrapolation Off Median Plane

From the vertical field $\vec{B}$ and derivatives in the median plane, first a transformation from polar to Cartesian coordinates is performed, following eqs (1.4.9 or 1.4.10), then, extrapolation off median plane is performed by means of Taylor expansions, following the procedure described in section 1.3.3.

Sharp Edge

Sharp edge field fall-off at a field boundary can only be simulated if the following conditions are fulfilled :

- entrance (resp. exit) field boundary coincides with entrance (resp. exit) dipole limit (it means in particular, see Fig. 11, $\omega^+ = ACENT$ (resp. $\omega^- = -(AT - ACENT)$), together with $\theta = 0$ at entrance (resp. exit) EFBs),

- analytical method for calculation of the mid-plane field derivatives is used.

DODECAPO : Dodecapole magnet (Fig. 24)

The meaning of parameters for *DODECAPO* is the same as for *QUADRUPO.*
In fringe field regions the magnetic field $\vec{B}(X,Y,Z)$ and its derivatives up to fourth order are derived from the scalar potential approximated to the 6th order in Y and Z

$$V(X,Y,Z) = G(X)\left(Y^4 - \frac{10}{3}Y^2Z^2 + Z^4\right)YZ$$
$$\text{with } G_0 = \frac{B_0}{R_0^5}$$

The modelling of the fringe field form factor $G(X)$ is described under *QUADRUPO*, p. 147.

Outside fringe field regions, or everywhere in sharp edge dodecapole ($\lambda_E = \lambda_S = 0$) , $\vec{B}(X,Y,Z)$ in the magnet is given by

$$B_X = 0$$
$$B_Y = G_0(5Y^4 - 10Y^2Z^2 + Z^4)Z$$
$$B_Z = G_0(Y^4 - 10Y^2Z^2 + 5Z^4)Y$$

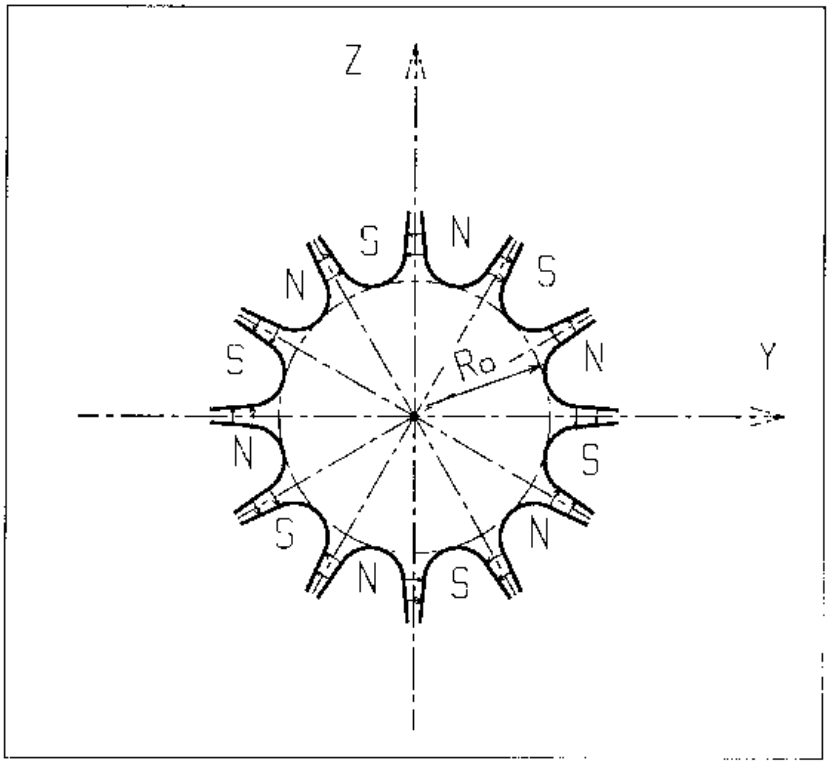


Figure 24: Dodecapole magnet

DRIFT or ESL : Field free drift space

DRIFT or *ESL* allow introduction of a drift space with length *XL* with positive or negative sign, anywhere in a structure. The associated equations of motion are (Fig. 25)

$$Y_2 = Y_1 + XL * \mathrm{tg}T$$
$$Z_2 = Z_1 + \frac{XL}{\cos T}\,\mathrm{tg}P$$
$$SAR_2 = SAR_1 + \frac{XL}{\cos T * \cos P}$$

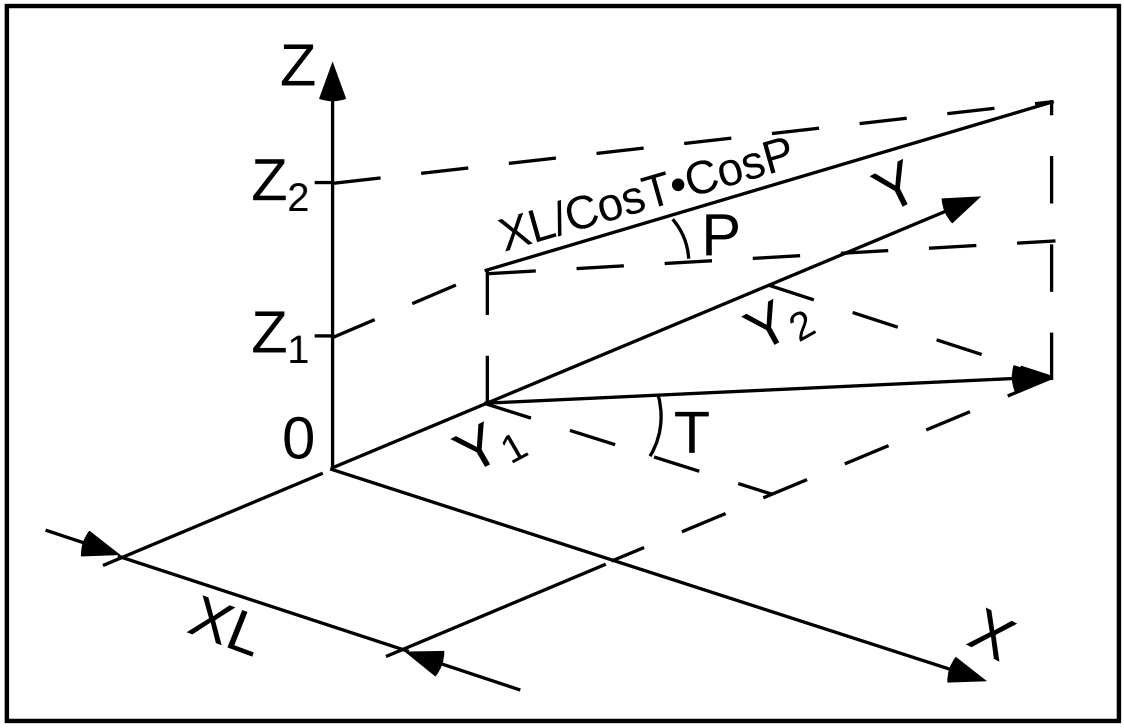


Figure 25: Transfer of particles in a drift space.

EBMULT : Electro-magnetic multipole

EBMULT simulates an electro-magnetic multipole, by addition of electric $(\vec{E})$ and magnetic $(\vec{B})$ multipole components (dipole to 20-pole). $\vec{E}$ and its derivatives $\frac{\partial^{i+j+k}\vec{E}}{\partial X^i \partial Y^j \partial Z^k}$ $(i+j+k \leq 4)$ are derived from the general expression of the multipole scalar potential (eq. 1.3.5), followed by a $\frac{\pi}{2n}$ rotation $(n = 1, 2, 3, ...)$ (see also *ELMULT*). $\vec{B}$ and its derivatives are derived from the same general potential, as described in section 1.3.7 (see also *MULTIPOL*).

The entrance and exit fringe fields of the $\vec{E}$ and $\vec{B}$ components are treated separately, in the same way as described under *ELMULT* and *MULTIPOL*, for each one of these two fields. Wedge angle correction is applied in sharp edge field model if $\vec{B}1$ is non zero, as in *MULTIPOL*. Any of the $\vec{E}$ or $\vec{B}$ multipole field component can be X-rotated independently of the others.

Use *PARTICUL* prior to *EBMULT*, for the definition of particle mass and charge.

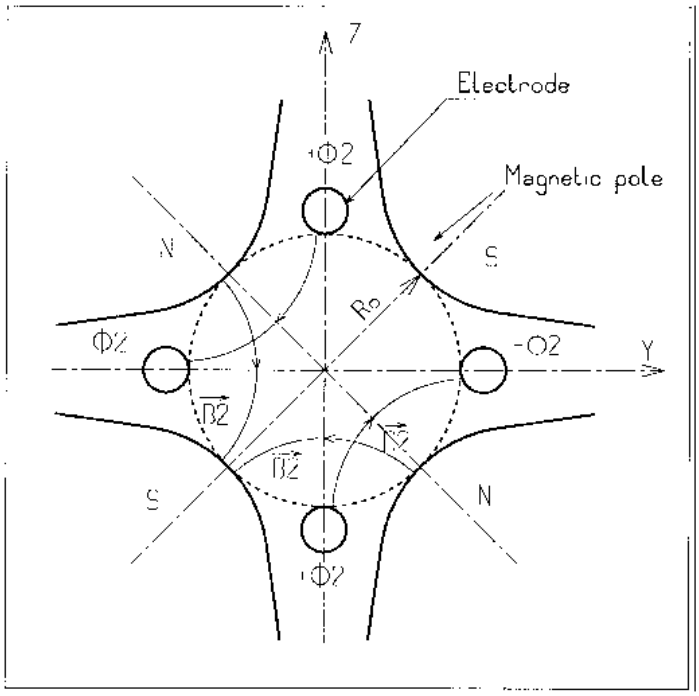


Figure 26: An example of $\vec{E}$, $\vec{B}$ multipole : the achromatic quadrupole (known for its allowing null second order chromatic aberrations [45]).

EL2TUB : Two-tube electrostatic lens

The lens is cylindrically symmetric about the X-axis.
The length and potential of the first (resp. second) electrode are $X1$ and $V1$ ($X2$ and $V2$). The distance between the two electrodes is D, and their inner radius is R_0 (Fig. 27). The model for the electrostatic potential along the axis is [46]

$$V(X) = \frac{V_2 - V_1}{2} \,\text{th}\, \frac{\omega x}{R_0} \left[+ \frac{V_1 + V_2}{2} \right] \qquad \text{if } D = 0$$

$$V(X) = \frac{V_2 - V_1}{2} \, \frac{1}{2\omega D / R_0} \, \ell n \, \frac{\text{ch}\, \omega \, \dfrac{x + D}{R_0}}{\text{ch}\, \omega \, \dfrac{x - D}{R_0}} \left[+ \frac{V_1 + V_2}{2} \right] \qquad \text{if } D \neq 0$$

(x = distance from half-way between the electrodes ; ω = 1.318 ; th = hyperbolic tangent ; ch = hyperbolic cosine) from which the field $\vec{E}(X, Y, Z)$ and its derivatives are derived following the procedure described in section 1.3.1 (note that they don't depend on the constant term $\left[\dfrac{V_1 + V_2}{2} \right]$ which disappears when differentiating).

Use *PARTICUL* prior to *EL2TUB*, for the definition of particle mass and charge.

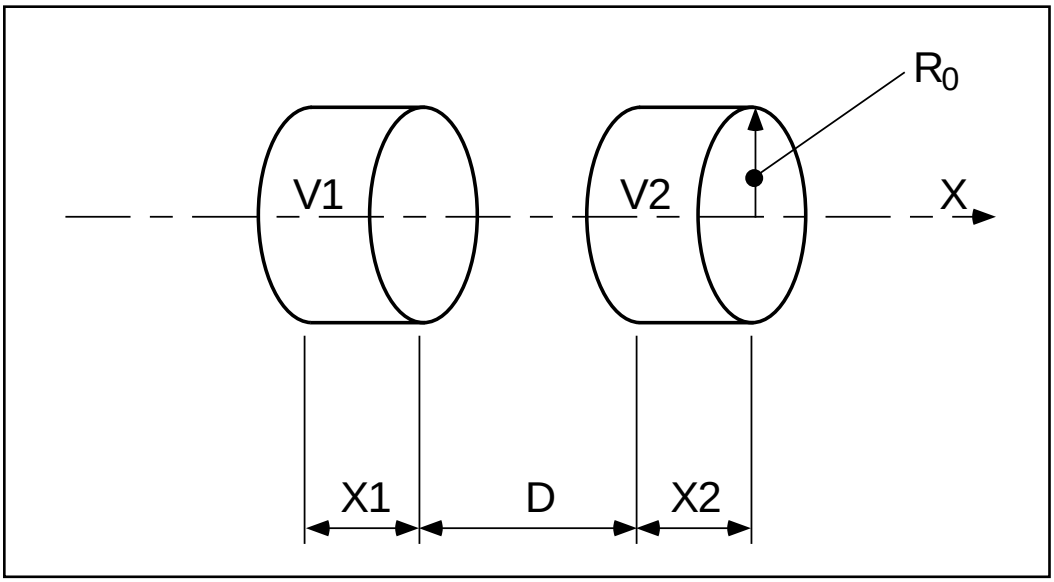


Figure 27: Two-electrode cylindrical electric lens.

ELMIR : Electrostatic N-electrode mirror/lens, straight slits

The device works as mirror or lens, horizontal or vertical. It is made of N 2-plate electrodes and has mid-plane symmetry.

Electrode lengths are $L1$, $L2$, ..., LN. D is the mirror/lens gap. The model for the Y-independent electrostatic potential is (after Ref. [47, p.412])

$$V(X,Z) = \sum_{i=2}^{N} \frac{Vi - Vi{-}1}{\pi} \arctan \frac{\sinh(\pi(X - Xi{-}1)/D)}{\cos(\pi Z/D)}$$

where Vi are the potential at the N electrodes (and normally $V1 = 0$ refers to the incident beam energy), Xi are the locations of the zero-length slits, X is the distance from the origin taken at the first slit (located at $X1 \equiv 0$ between the first and second electrodes). From $V(X,Z)$ the field $\vec{E}(X,Y,Z)$ and derivatives are deduced following the procedure described in section 1.3.7 (page 27).

The total X-extent of the mirror/lens is $L = \sum_{i=1}^{N} Li$.

In the mirror mode (option *MT* $= 11$ for vertical mid-plane or *MT* $= 12$ for horizontal mid-plane) stepwise integration starts at $X = -L1$ (entrance of the first electrode) and terminates either when back to $X = -L1$ or when reaching $X = L - L1$ (end of the $N - th$ electrode). In the latter case particles are stopped with their index IEX set to -8 (see section 5.6.11 on page 184). Normally $X1$ should exceed $3D$ (enough that $V(X < X1)$ have negligible effect in terms of trajectory behavior).

In the lens mode (option flag *MT* $= 21$ for vertical mid-plane or *MT* $= 22$ for horizontal mid-plane) stepwise integration starts at $X = -L1$ (entrance of the first electrode) and terminates either when reaching $X = L - L1$ (end of the $N - th$ electrode) or when the particle deflection exceeds $\pi/2$. In the latter case the particle is stopped with their index IEX set to -3.

Use *PARTICUL* prior to *ELMIR*, for the definition of particle mass and charge.

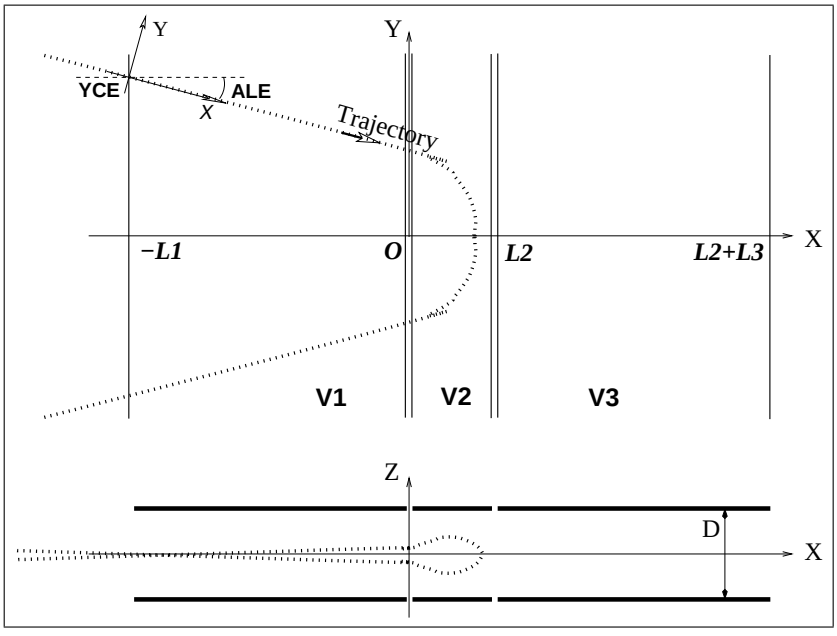


Figure 28: Electrostatic N-electrode mirror/lens, straight slits, in the case $N = 3$, in horizontal mirror mode ($MT = 11$).
Possible non-zero entrance quantities YCE, ALE should be specified using *CHANGREF*, or using *KPOS*=3 with *YCE* and *ALE*=half-deviation matched to the reference trajectory.

ELMIRC : Electrostatic N-electrode mirror/lens, circular slits [47]

The device works as mirror or lens, horizontal or vertical. It is made of N 2-plate electrodes and has mid-plane symmetry[6].

Electrode slits are circular, concentric with radii $R1$, $R2$, ..., $R_{N\text{-}1}$, D is the mirror/lens gap. The model for the mid-plane ($Z = 0$) radial electrostatic potential is (after Ref. [47, p.443])

$$V(r) = \sum_{i=2}^{N} \frac{Vi - Vi{-}1}{\pi} \arctan\left(\sinh \frac{\pi(r - Ri{-}1)}{D}\right)$$

where Vi are the potential at the N electrodes (and normally $V1 = 0$ refers to the incident beam energy). r is the current radius.

The mid-plane field $\vec{E}(r)$ and its r-derivatives are first derived by differentiation, then $\vec{E}(r, Z)$ and derivatives are obtained from Taylor expansions and Maxwell relations. Eventually a transformation to the rotating frame provides $\vec{E}(X, Y, Z)$ and derivatives as involved in eq. 1.2.15.
Stepwise integration starts at entrance (defined by RE, TE) of the first electrode and terminates when rotation of the reference rotating frame (RM, X, Y) has reached the value AT. Normally, $R1 - RE$ and $R1 - RS$ should both exceed $3D$ (so that potential tails have negligible effect in terms of trajectory behavior).

Positioning of the element is performed by means of *KPOS* (see section 5.6.8).

Use *PARTICUL* prior to *ELMIRC*, for the definition of particle mass and charge.

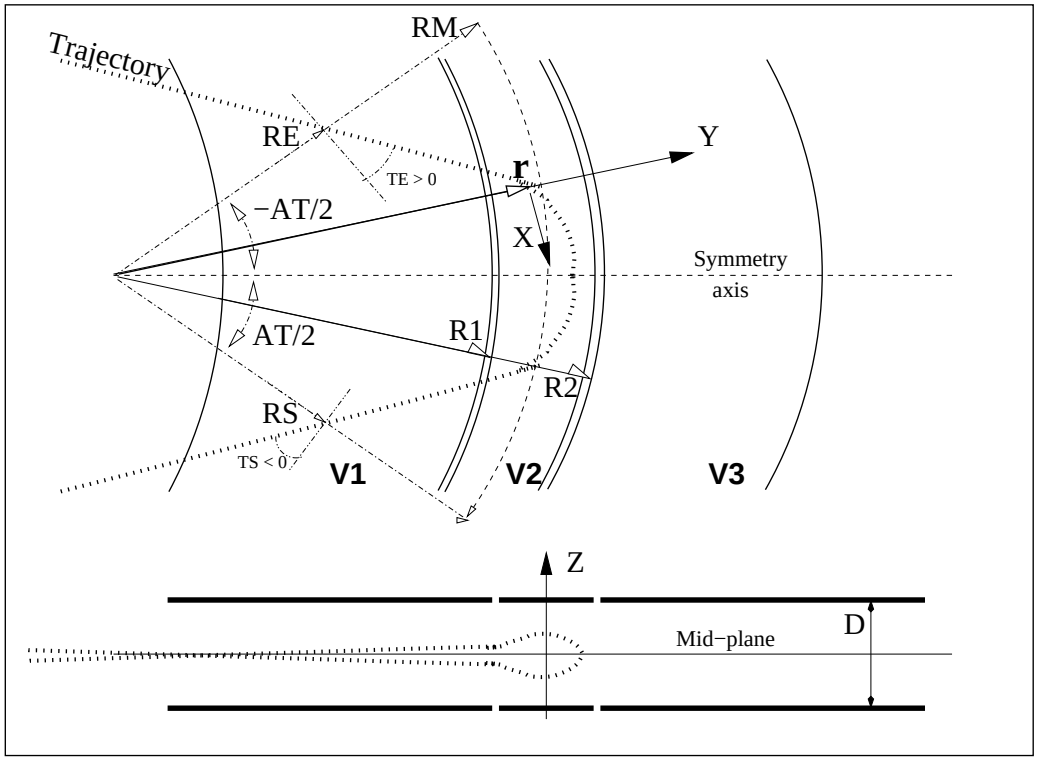


Figure 29: Electrostatic N-electrode mirror/lens, circular slits, in the case $N = 3$, in horizontal mirror mode.

[6]NOTE : in the present version of the code, the sole horizontal mirror mode is operational, and N is limited to 3.

ELMULT : Electric multipole

The simulation of multipolar electric field $\vec{M}_E$ proceeds by addition of the dipolar $(\vec{E}1)$, quadrupolar ($\vec{E}2$), sextupolar ($\vec{E}3$), etc., up to 20-polar $(\vec{E}10)$ components, and of their derivatives up to fourth order, following

$$\begin{aligned}
\vec{M}_E &= \vec{E}1 + \vec{E}2 + \vec{E}3 + \ ... \ + \vec{E}10 \\
\frac{\partial \vec{M}_E}{\partial X} &= \frac{\partial \vec{E}1}{\partial X} + \frac{\partial \vec{E}2}{\partial X} + \frac{\partial \vec{E}3}{\partial X} + \ ... \ + \frac{\partial \vec{E}10}{\partial X} \\
\frac{\partial^2 M_E}{\partial X \partial Z} &= \frac{\partial^2 \vec{E}1}{\partial X \partial Z} + \frac{\partial^2 \vec{E}2}{\partial X \partial Z} + \frac{\partial^2 \vec{E}3}{\partial X \partial Z} + \ ... \ + \frac{\partial^2 \vec{E}10}{\partial X \partial Z} \\
&\text{etc.}
\end{aligned}$$

The independent components $\vec{E}1$ to $\vec{E}10$ and their derivatives up to the fourth order are calculated by differentiating the general multipole potential given in eq. 1.3.5 (page 28), followed by a $\frac{\pi}{2n}$ rotation about the X-axis, so that the so defined right electric multipole of order n, and of strength [45, 48]

$$K_n = \frac{1}{2} \frac{\gamma}{\gamma^2 - 1} \frac{V_n}{R_0^n}$$

(V_n = potential at the electrode, R_0 = radius at pole tip, γ = relativistic Lorentz factor of the particle) has the same focusing effect as the right magnetic multipole of order n and strength $K_n = \frac{B_n}{R_0^{n-1} B\rho}$ (B_n = field at pole tip, $B\rho$ = particle rigidity, see *MULTIPOL*).

The entrance and exit fringe fields are treated separately. They are characterized by the integration zone X_E at entrance and X_S at exit, as for *QUADRUPO*, and by the extent λ_E at entrance, λ_S at exit. The fringe field extents for the dipole component are λ_E and λ_S. The fringe field extent for the quadrupolar (sextupolar, ..., 20-polar) component is given by a coefficient E_2 (E_3, ..., E_{10}) at entrance, and S_2 (S_3, ..., S_{10}) at exit, such that the fringe field extent is $\lambda_E * E_2$ ($\lambda_E * E_3$, ..., $\lambda_E * E_{10}$) at entrance and $\lambda_S * S_2$ ($\lambda_S * S_3$, ..., $\lambda_S * S_{10}$) at exit.

If $\lambda_E = 0$ ($\lambda_S = 0$) the multipole lens is considered to have a sharp edge field at entrance (exit), and then, X_E (X_S) is forced to zero (for the mere purpose of saving computing time).

If $E_i = 0$ ($S_i = 0$) ($i = 2, 10$), the entrance (exit) fringe field for multipole component i is considered as a sharp edge field.

Any multipole component $\vec{E}i$ can be rotated independently by an angle RXi around the longitudinal X-axis, for the simulation of positioning defects, as well as skew lenses.

Use *PARTICUL* prior to *ELMULT*, for the definition of particle mass and charge.

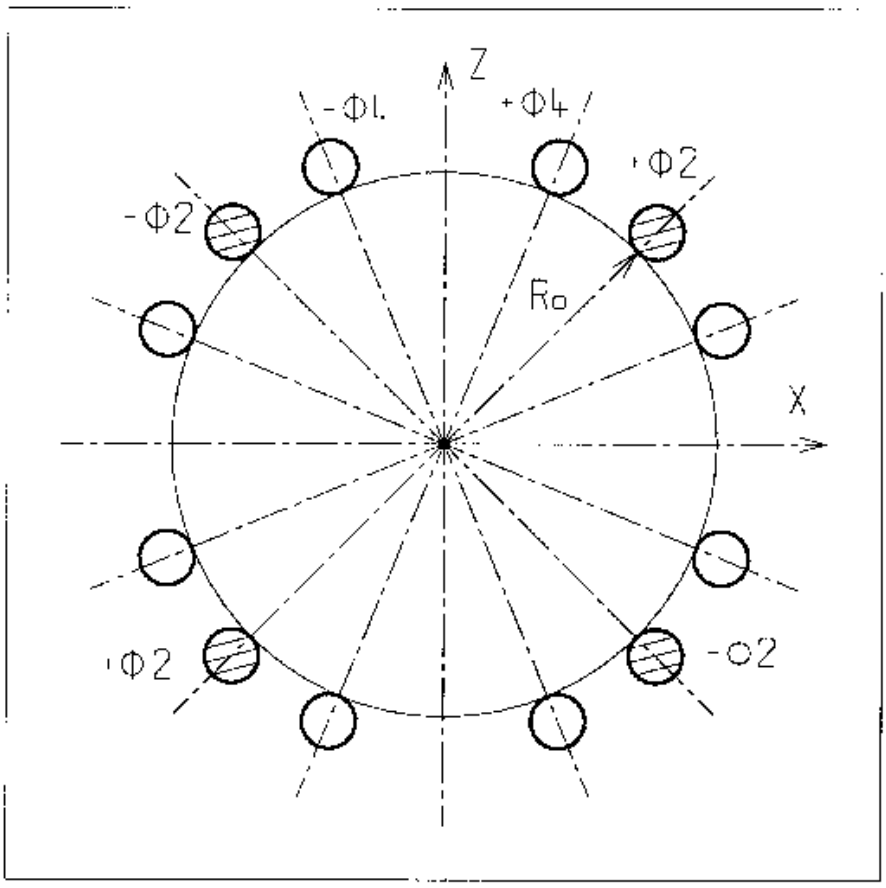


Figure 30: An electric multipole combining skew-quadrupole ($\vec{E}2 \neq \vec{0},\ \vec{R}2 = \pi/4$) and skew-octupole ($\vec{E}4 \neq \vec{0},\ \vec{R}4 = \pi/8$) components ($\vec{E}1 = \vec{E}3 = \vec{E}5 = \ ... \ = \vec{E}10 = \vec{0}$) [48].

ELREVOL : 1-D uniform mesh electric field map

ELREVOL reads a 1-D axial field map from a storage data file, whose content must fit the following *FORTRAN* reading sequence

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
DO 1 I=1, IX
   IF (BINARY) THEN
      READ(NL) X(I), EX(I)
   ELSE
      READ(NL,*) X(I), EX(I)
   ENDIF
  1     CONTINUE
```

where IX is the number of nodes along the (symmetry) X-axis, $X(I)$ their coordinates, and $EX(I)$ are the values of the X component of the field. EX is normalized with *ENORM* prior to ray-tracing. As well the longitudinal coordinate X is normalized with a *XNORM* coefficient (useful to convert to centimeters, the working units in **zgoubi**).

X-cylindrical symmetry is assumed, resulting in EY and EZ taken to be zero on axis. $\vec{E}(X,Y,Z)$ and its derivatives along a particle trajectory are calculated by means of a 5-points polynomial interpolation followed by second order off-axis extrapolation (see sections 1.3.1 and 1.4.1).

Entrance and/or exit integration boundaries may be defined in the same way as in *CARTEMES* by means of the flag ID and coefficients A, B, C, A', B', C'.

Use *PARTICUL* prior to *ELREVOL*, for the definition of particle mass and charge.

EMMA : 2-D Cartesian or cylindrical mesh field map for EMMA FFAG

EMMA is dedicated to the reading and treatment of 2-D or 3-D Cartesian mesh field maps representing the EMMA FFAG cell quadrupole doublet[7] [51, 52].

EMMA can sum up independent field maps of each of the two quadrupoles, with each its scaling coefficient. The two maps can be radially positioned independently of one another at Y_F, Y_D respectively, just like the actual *EMMA* quadrupoles. In particular,

MOD : operational and map *FORMAT* reading mode ;
MOD$\leq$19 : Cartesian mesh ;
MOD$\geq$20 : cylindrical mesh.

MOD=0 : two 2D maps, one representing QF, one reprensenting QD. A single map, superimposition of both, is built prior to tracking and used for tracking.

MOD=1 : two 2D maps, one representing QF, one reprensenting QD, a resulting single map is devised in the following way : QF_new is interpolated from QF with dr=xF, QD_new is interpolated from QD with dr=xD. A single map, superimposition of both, is built prior to tracking and used for tracking.

The parameters that move/position the maps, as (Y_F, Y_D), are accessible from the FIT, allowing to adjust the cell tunes.

EMMA works much like *TOSCA*. Refer to that keyword, and to the *FORTRAN* file `emmac.f`, for details.

[7] The stepwise ray-tracing code Zgoubi is the on-line model code for the worlds first non-scaling FFAG experiment.

FFAG : FFAG magnet, N-tuple [42, 43]

FFAG works much like *DIPOLES* as to the field modelling, apart from the radial dependence of the field, $B = B_0(r/r_0)^k$, so-called "scaling". Note that *DIPOLES* does similar job by using a Taylor r-expansion of $B_0(r/r_0)^k$.

The *FFAG* procedure allows overlapping of fringe fields of neighboring dipoles, thus simulating in some sort the field in a dipole N-tuple - as for instance in an FFAG doublet or triplet. A detailed application, with five dipoles, can be found in Ref. [42]. This is done in the way described below.

The dimensioning of the magnet is defined by

AT : total angular aperture
RM : mean radius used for the positioning of field boundaries

For each one of the $N = 1$ to (maximum) 5 dipoles of the N-tuple, the two effective field boundaries (entrance and exit EFBs) from which the dipole field is drawn are defined from geometric boundaries, the shape and position of which are determined by the following parameters (in the same manner as in *DIPOLE*, *DIPOLE-M*) (see Fig. 11-A page 92, and Fig. 31)

ACN_i : arbitrary inner angle, used for EFBs positioning
ω : azimuth of an EFB with respect to *ACN*
θ : angle of an EFB with respect to its azimuth (wedge angle)
R_1, R_2 : radius of curvature of an EFB
U_1, U_2 : extent of the linear part of an EFB

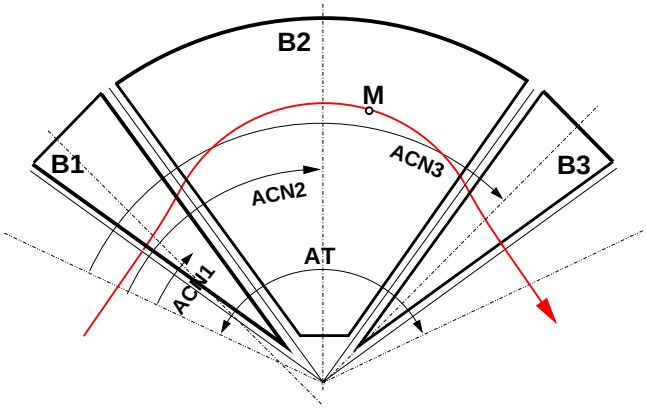


Figure 31: Definition of a dipole N-tuple ($N = 3$, a triplet here) using the *DIPOLES* or *FFAG* procedures.

Calculation of the Field From a Single Dipole

The magnetic field is calculated in polar coordinates. At all (R, θ) in the median plane ($z = 0$), the magnetic field due a single one (index i) of the dipoles of a N-tuple FFAG magnet is written

$$B_{Zi}(R, \theta) = B_{Z0,i}\, \mathcal{F}_i(R, \theta)\ (R/R_M)^{K_i}$$

wherein $B_{Z0,i}$ is a reference field, at reference radius RM_i, whereas $\mathcal{F}(R, \theta)$ is calculated as described below.

Calculation of $\mathcal{F}_i(R, \theta)$

The fringe field coefficient $\mathcal{F}_i(R, \theta)$ associated with a dipole is computed as in the procedure *DIPOLES* (eq. 5.4.16), including (rigorously if the interpolation method is used, see page 119, or to order zero if the analytic method is used, see page 120) radial dependence of the gap size

$$g(R) = g_0 \, (RM/R)^{\kappa} \tag{5.4.19}$$

so to simulate the effect of gap shaping on $B_{Zi}(R,\theta)|_R$ field fall-off, over the all radial extent of a scaling FFAG dipole (with normally - but not necessarily in practice - $\kappa \approx K_i$).

Calculation of the Field Resulting From All N Dipoles

For the rest, namely, calculation of the full field at particle position from the N dipoles, analytical calculation or numerical interpolation of the mid-plane field derivatives, extrapolation off median plane, etc., things are performed exactly as in the case of the *DIPOLES* procedure (see page 119).

Sharp Edge

Sharp edge field fall-off at a field boundary can only be simulated if the following conditions are fulfilled :

- entrance (resp. exit) field boundary coincides with entrance (resp. exit) dipole limit (it means in particular, see Fig. 11, $\omega^+ = ACENT$ (resp. $\omega^- = -(AT - ACENT)$), together with $\theta = 0$ at entrance (resp. exit) EFBs),
- analytical method for calculation of the mid-plane field derivatives is used.

FFAG-SPI : Spiral FFAG magnet, N-tuple [43, 49]

FFAG-SPI works much like *FFAG* as to the field modelling, with essentially a different axial dependence.

The *FFAG-SPI* procedure allows overlapping of fringe fields of neighboring dipoles, thus simulating in some sort the field in a dipole N-tuple (similar to Fig. 31, page 132). This allows for instance accounting for fringe field effects, or clamps, as schemed in Fig. 32.

The dimensioning of the magnet is defined by

AT : total angular aperture
RM : mean radius used for the positioning of field boundaries

For each one of the $N = 1$ to (maximum) 5 dipoles of the N-tuple, the two effective field boundaries (entrance and exit EFBs) from which the dipole field is drawn are defined from geometric boundaries, the shape and position of which are determined by the following parameters

ACN_i : arbitrary inner angle, used for EFBs positioning
ω : azimuth of an EFB with respect to *ACN*
ξ : spiral angle

with ACN_i and ω as defined in Fig. 32 (similar to what can be found in Figs. 31 and 11-A).

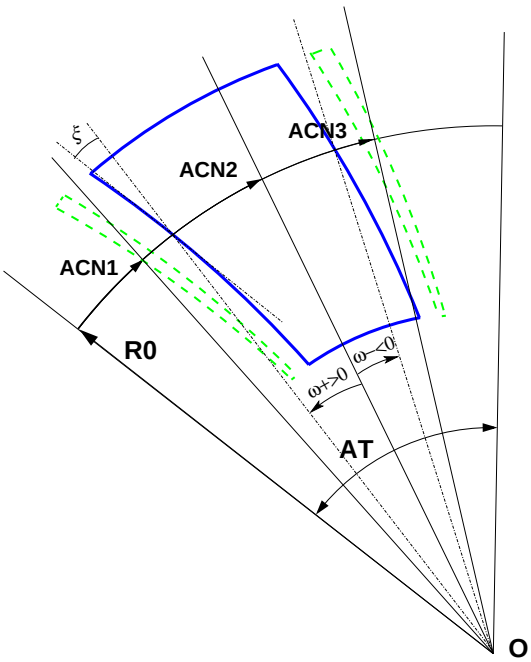


Figure 32: A N-tuple spiral sector FFAG magnet ($N = 3$ here, simulating active field clamps at entrance and exit side of a central dipole).

Calculation of the Field From a Single Dipole

The magnetic field is calculated in polar coordinates. At all (R, θ) in the median plane ($Z = 0$), the magnetic field due a single one (index i) of the dipoles of a N-tuple spiral FFAG magnet is written

$$B_{Zi}(R,\theta) = B_{Z0,i}\,\mathcal{F}_i(R,\theta)\;(R/R_M)^{K_i}$$

wherein $B_{Z0,i}$ is a reference field, at reference radius RM_i, whereas $\mathcal{F}(R,\theta)$ is calculated as described below.

Calculation of $\mathcal{F}_i(R,\theta)$

The fringe field coefficient $\mathcal{F}_i(R,\theta)$ associated with a dipole is computed as in the procedure *DIPOLES* (eq. 5.4.16), including radial dependence of the gap size

$$g(R) = g_0\,(RM/R)^{\kappa} \tag{5.4.20}$$

so to simulate the effect of gap shaping on $B_{Zi}(R,\theta)|_R$ field fall-off, over the all radial extent of the dipole (with normally - yet not necessarily in practice - $\kappa \approx K_i$).

Calculation of the Full Field From All N Dipoles

For the rest, namely calculation of the full field at particle position, as resulting from the N dipoles, calculation of the mid-plane field derivatives, extrapolation off median plane, etc., things are performed in the same manner as for the *DIPOLES* procedure (see page 119).

GOTO : Branching statement [53]

******** To be completed **********
GOTO is a conditional goto, working much like the fortran branching statement. It sends the execution pointer to some keyword (*e.g.*, "*MARKER*") in the optical sequence, characterized by one or two specific labels.

Available options :

- pass number

GOTO accepts *GOBACK* as an argument, this sends the execution pointer back to the next keyword immediately following the calling *GOTO.*

GOTO/GOBACK for instance can be used in RLA type of structures, to direct the **zgoubi** amongst the series of spreaders of mergers. A example is given in ********.

An example of an input data file of a recirculator ring, structured with *GOTO* statements, can be found in page 243.

INCLUDE : File include statement

INCLUDE works much like the fortran “include” statement : it allows a fragment of an input data list to be taken from a separate file ; when *INCLUDE* is met in the input data file, the content of the separate file, or part of it, is inserted in the data sequence.
The INCLUDE statement takes the following arguments :

- the number of input files to be included,
- their names, in as many lines, one name per line.

The file name may be followed by *’[LBL_1A,LBL_2A:LBL_1B,LBL_2B]’*, with any of these four labels optional, as follows :

- LBL_1A [LBL_2A] identify to the first (and possibly second) label of some keyword in the include file, from which the include process will start (whatever precedes will be ignored). If none of *’LBL_1A, _2A’* is declared, the include process starts from the begining of the include file. Declaring the second label only takes the form *’[,LBL_2A:LBL_1B,LBL_2B]’*.
- LBL_1B [LBL_2B] identify to the first (and second) label of some keyword in the include file, at which the include process will end (whatever follows will be ignored). If none of *’LBL_1B, _2B’* is declared, the include file will be included down to its end. Declaring the second label only takes the form *’[LBL_1A,LBL_2A: ,LBL_2B]’*.

Because **zgoubi** uses the reserved name zgoubi.dat as its standard working data file name, and as the presence of *INCLUDE* will cause the input data list to be changed (it is expanded to include the content of all included files), the user’s input data file is no longer allowed the name zgoubi.dat if it contains *INCLUDE* keyword(s) (zgoubi.dat is reserved as the name of the expanded input file to zgoubi). This is overcome by using the command ’zgoubi -fileIn myFile’ instead, with myFile≠zgoubi.dat (see Sec. 1.2 in PART D of the guide, page 321).

An *INCLUDE* file may contain one or more *INCLUDE*. There is a limit in the allowed depth of *INCLUDE* statements within *INCLUDE* files, however high. **zgoubi** will warn the user when it is exceeded. It can be changed in prdata.f.

By contrast with other keywords, *INCLUDE* is not part on the LSTKEY.H keyword list (found in the *’include’* folder). Instead, possible presence of *INCLUDE* in the input data file is dealt with by prdata program (prdata.f source file) in a simple way : the input data file is expanded so eliminating all *INCLUDE* occurences, and the expanded, complete, data list is stored in the standard **zgoubi** input data file zgoubi.dat. Thus **zgoubi** eventually actually works using a regular, complete, data list in a single input file, zgoubi.dat.

An example of an input data file of a recirculating ring, structured with *INCLUDE* statements, can be found in page 243.

MAP2D : 2-D Cartesian uniform mesh field map - arbitrary magnetic field [53]

MAP2D reads a 2-D field map that provides the three components B_X, B_Y, B_Z of the magnetic field at all nodes of a 2-D Cartesian uniform mesh in an (X, Y) plane. No particular symmetry is assumed, which allows the treatment of any type of field (*e.g.*, solenoidal, or dipole, helical dipole, at arbitrary Z elevation - the map needs not be a mid-plane map).

The field map data file has to be be filled with a format that satisfies the *FORTRAN* reading sequence below (in principle compatible with *TOSCA* code outputs), details and possible updates are to be found in the source file `'fmapw.f'` :

```
      OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
       DO 1 J=1,JY
          DO 1 I=1,IX
             IF (BINARY) THEN
                READ(NL) Y(J), Z, X(I), BY(I,J), BZ(I,J), BX(I,J)
             ELSE
                READ(NL,100) Y(J), Z, X(I), BY(I,J), BZ(I,J), BX(I,J)
100             FORMAT (1X, 6E11.4)
             ENDIF
      1       CONTINUE
```

IX (JY) is the number of longitudinal (transverse horizontal) nodes of the 2-D uniform mesh, Z is the considered Z-elevation of the map. For binary files, FNAME must begin with 'B_ ' or 'b_', a flag 'BINARY' will thus be set to '.TRUE.'. The field $\vec{B} = (B_X, B_Y, B_Z)$ is next normalized with BNORM, prior to ray-tracing. As well the coordinates X, Y are normalized with *X-, Y-NORM* coefficients (useful to convert to centimeters, the working units in **zgoubi**).

At each step of the trajectory of a particle, the field and its derivatives are calculated using a second or fourth degree polynomial interpolation followed by a Z extrapolation (see sections 1.3.4 page 27, 1.4.3 page 31). The interpolation grid is 3*3-node for 2nd order (option *IORDRE = 2*) or 5*5 for 4th order (option *IORDRE = 4*).

Entrance and/or exit integration boundaries may be defined, in the same way as for *CARTEMES*.

MAP2D-E : 2-D Cartesian uniform mesh field map - arbitrary electric field

MAP2D-E reads a 2-D field map that provides the three components E_X, E_Y, E_Z of the electric field at all nodes of a 2-D Cartesian uniform mesh in an (X, Y) plane. No particular symmetry is assumed, which allows the treatment of any type of field (*e.g.*, field of a parallel-plate mirror with arbitrary Z elevation - the map needs not be a mid-plane map).

The field map data file has to be be filled with a format that satisfies the *FORTRAN* reading sequence below (in principle compatible with *TOSCA* code outputs), details and possible updates are to be found in the source file `'fmapw.f'` :

```
      OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
       DO 1 J=1,JY
          DO 1 I=1,IX
             IF (BINARY) THEN
                READ(NL) Y(J), Z, X(I), EY(I,J), EZ(I,J), EX(I,J)
             ELSE
                READ(NL,100) Y(J), Z, X(I), EY(I,J), EZ(I,J), EX(I,J)
100             FORMAT (1X, 6E11.4)
             ENDIF
      1       CONTINUE
```

IX (JY) is the number of longitudinal (transverse horizontal) nodes of the 2-D uniform mesh, Z is the considered Z-elevation of the map. For binary files, FNAME must begin with 'E_' or 'b_', a flag 'BINARY' will thus be set to '.TRUE.'. The field $\vec{E} = (E_X, E_Y, E_Z)$ is next normalized with ENORM, prior to ray-tracing. As well the coordinates X, Y are normalized with *X-,Y-NORM* coefficients (useful to convert to centimeters, the working units in **zgoubi**.

At each step of the trajectory of a particle, the field and its derivatives are calculated using a second or fourth degree polynomial interpolation followed by a Z extrapolation (see sections 1.3.4 page 27, 1.4.3 page 31). The interpolation grid is 3*3-node for 2nd order (option *IORDRE = 2*) or 5*5 for 4th order (option *IORDRE = 4*).

Entrance and/or exit integration boundaries may be defined, in the same way as for *CARTEMES*.

MARKER : Marker

MARKER does nothing. Just a marker. No data.

As any other keyword, *MARKER* is allowed two *LABEL*s. Using '.plt' as a second *LABEL* will cause storage of current coordinates into zgoubi.plt.

MULTIPOL : Magnetic multipole

The simulation of multipolar magnetic field $\vec{M}$ by *MULTIPOL* proceeds by addition of the dipolar ($\vec{B}1$), quadrupolar ($\vec{B}2$), sextupolar ($\vec{B}3$), etc., up to 20-polar ($\vec{B}10$) components, and of their derivatives up to fourth order, following

$$\vec{M} = \vec{B}1 + \vec{B}2 + \vec{B}3 + \ ... \ + \vec{B}10$$
$$\frac{\partial \vec{M}}{\partial X} = \frac{\partial \vec{B}1}{\partial X} + \frac{\partial \vec{B}2}{\partial X} + \frac{\partial \vec{B}3}{\partial X} + \ ... \ + \frac{\partial \vec{B}10}{\partial X}$$
$$\frac{\partial^2 \vec{M}}{\partial X \partial Z} = \frac{\partial^2 \vec{B}1}{\partial X \partial Z} + \frac{\partial^2 \vec{B}2}{\partial X \partial Z} + \frac{\partial^2 \vec{B}3}{\partial X \partial Z} + \ ... \ + \frac{\partial^2 \vec{B}10}{\partial X \partial Z}$$
etc.

The independent components $\vec{B}1$, $\vec{B}2$, $\vec{B}3$, ..., $\vec{B}10$ and their derivatives up to the fourth order are calculated as described in section 1.3.7.

The entrance and exit fringe fields are treated separately. They are characterized by the integration zone X_E at entrance and X_S at exit, as for *QUADRUPO*, and by the extent λ_E at entrance, λ_S at exit. The fringe field extents for the dipole component are λ_E and λ_S. The fringe field for the quadrupolar (sextupolar, ..., 20-polar) component is given by a coefficient E_2 (E_3, ..., E_{10}) at entrance, and S_2 (S_3, ..., S_{10}) at exit, such that the extent is $\lambda_E * E_2$ ($\lambda_E * E_3$, ..., $\lambda_E * E_{10}$) at entrance and $\lambda_S * S_2$ ($\lambda_S * S_3$, ..., $\lambda_S * S_{10}$) at exit.

If $\lambda_E = 0$ ($\lambda_S = 0$) the multipole lens is considered to have a sharp edge field at entrance (exit), and then, X_E (X_S) is forced to zero (for the mere purpose of saving computing time). If $E_i = 0$ ($S_i = 0$) ($i = 2,\ 10$), the entrance (exit) fringe field for the multipole component i is considered as a sharp edge field. In sharp edge field model, the wedge angle vertical first order focusing effect (if $\vec{B}1$ is non zero) is simulated at magnet entrance and exit by a kick $P_2 = P_1 - Z_1 \tan(\epsilon/\rho)$ applied to each particle (P_1, P_2 are the vertical angles upstream and downstream of the EFB, Z_1 is the vertical particle position at the EFB, ρ the local horizontal bending radius and ϵ the wedge angle experienced by the particle ; ϵ depends on the horizontal angle T).

Any multipole component $\vec{B}i$ can be rotated independently by an angle RXi around the longitudinal X-axis, for the simulation of positioning defects, as well as skew lenses.

Magnet (mis-)alignment is assured by *KPOS*. *KPOS* also allows some degrees of automatic alignment useful for periodic structures (section 5.6.8).

OCTUPOLE : Octupole magnet (Fig. 33)

The meaning of parameters for *OCTUPOLE* is the same as for *QUADRUPO*. In fringe field regions the magnetic field $\vec{B}(X,Y,Z)$ and its derivatives up to fourth order are derived from the scalar potential approximated to the 8-th order in Y and Z

$$V(X,Y,Z) = \left(G(X) - \frac{G''(X)}{20}\,(Y^2+Z^2) + \frac{G''''(X)}{960}\,(Y^2+Z^2)^2 \right)(Y^3Z - YZ^3)$$

$$\text{with } G_0 = \frac{B_0}{R_0^3}$$

The modelling of the fringe field form factor $G(X)$ is described under *QUADRUPO*, p. 147.

Outside fringe field regions, or everywhere in sharp edge dodecapole ($\lambda_E = \lambda_S = 0$) , $\vec{B}(X,Y,Z)$ in the magnet is given by

$$B_X = 0$$
$$B_Y = G_0(3Y^2 - Z^2)\,Z$$
$$B_Z = G_0(Y^2 - 3Z^2)\,Y$$

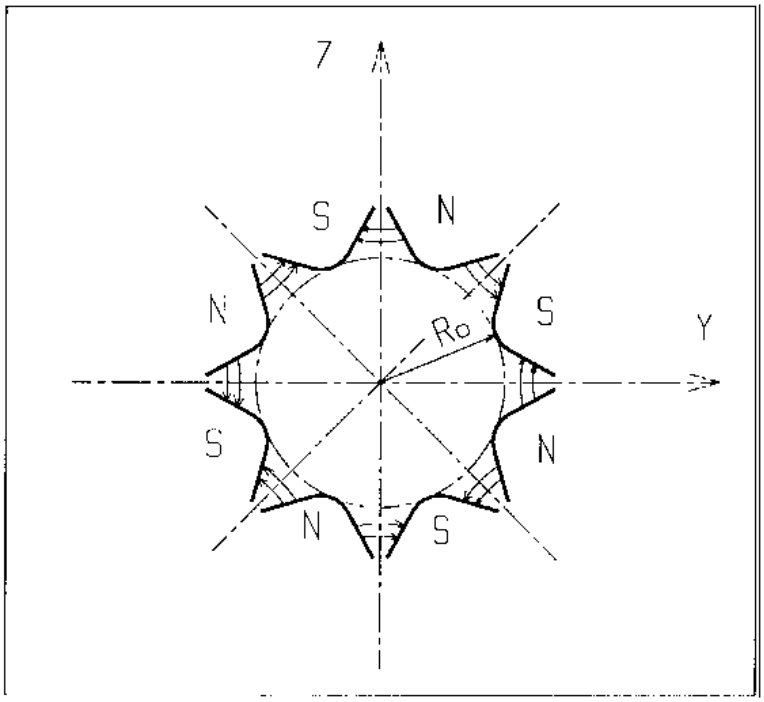


Figure 33: Octupole magnet

POISSON :Read magnetic field data from *POISSON* output

This keyword allows reading a field profile $B(X)$ from *POISSON* output. Let *FNAME* be the name of this output file (normally, *FNAME* = outpoi.lis) ; the data are read following the *FORTRAN* statements here under

```
I = 0
  11    CONTINUE
  I = I + 1
          READ(LUN,101,ERR=10,END=10) K, K, K, R, X(I), R, R, B(I)
  101     FORMAT(I1, I3, I4, E15.6, 2F11.5, 2F12.3)
        GOTO 11
  10    CONTINUE
        ...
```

where X(I) is the longitudinal coordinate, and B(I) is the Z component of the field at a node (I) of the mesh. Ks and Rs are dummy variables appearing in the *POISSON* output file outpoi.lis but not used here.

From this field profile, a 2-D median plane map is built, with a rectangular and uniform mesh ; mid-plane symmetry is assumed. The field at each node (X_i, Y_j) of the map is $B(X_i)$, independent of Y_j (*i.e.*, the distribution is uniform in the Y direction).

For the rest, *POISSON* works in a way similar to *CARTEMES.*

POLARMES : 2-D polar mesh magnetic field map

Similar to *CARTEMES*, apart from the polar mesh frame : IX is the number of angular nodes, JY the number of radial nodes ; $X(I)$ and $Y(J)$ are respectively the angle and radius of a node (these parameters are similar to those entering in the definition of the field map in *DIPOLE-M*).

PS170 : Simulation of a round shape dipole magnet

PS170 is dedicated to a 'rough' simulation of CERN *PS170* spectrometer dipole.

The field B_0 is constant inside the magnet, and zero outside. The pole is a circle of radius R_0, centered on the X axis. The output coordinates are generated at the distance *XL* from the entrance (Fig. 34).

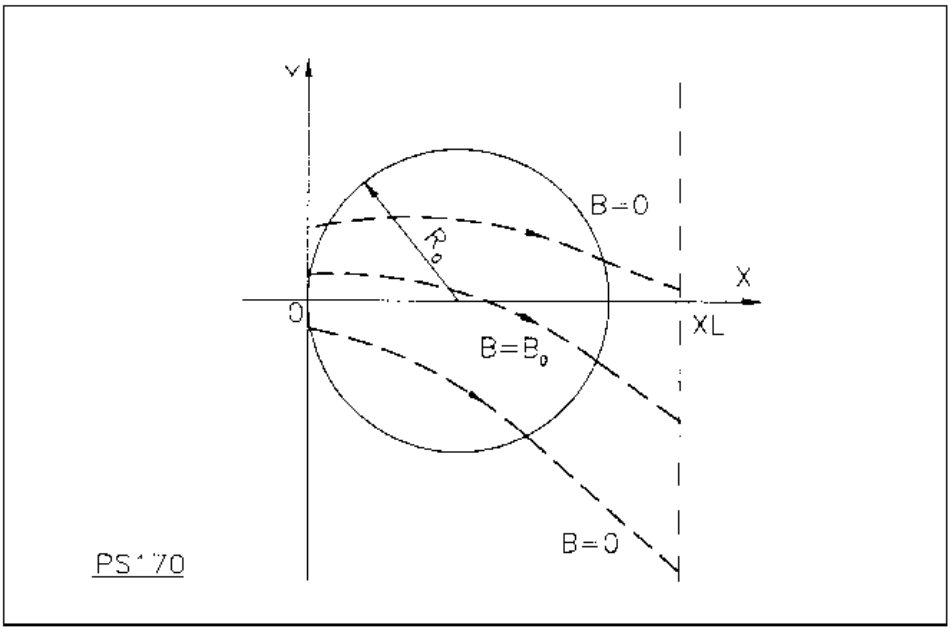


Figure 34: Scheme of the PS170 magnet simulation.

QUADISEX, SEXQUAD : Sharp edge magnetic multipoles

SEXQUAD defines in a simple way a sharp edge field with quadrupolar, sextupolar and octupolar components. *QUADISEX* adds a dipole component. The length of the element is *XL*. The vertical component $B \equiv B_Z(X, Y, Z = 0)$ of the field and its derivatives in median plane are calculated at each step using the following expressions

$$
\begin{aligned}
B &= B_0 \left(U + \frac{N}{R_0} Y + \frac{B}{R_0^2} Y^2 + \frac{G}{R_0^3} Y^3 \right) \\
\frac{\partial B}{\partial Y} &= B_0 \left(\frac{N}{R_0} + 2\frac{B}{R_0^2} Y + 3\frac{G}{R_0^3} Y^2 \right) \\
\frac{\partial^2 B}{\partial Y^2} &= B_0 \left(2\frac{B}{R_0^2} + 6\frac{G}{R_0^3} Y \right) \\
\frac{\partial^3 B}{\partial Y^3} &= 6 B_0 \frac{G}{R_0^3}
\end{aligned}
$$

and then extrapolated out of the median plane by Taylor expansion in Z (see section 1.3.3).

With option *SEXQUAD*, $U = 0$, while with *QUADISEX*, $U = 1$.

QUADRUPO : Quadrupole magnet (Fig. 35)

The length of the magnet *XL* is the distance between the effective field boundaries (EFB), Fig. 36. The field at the pole tip R_0 is B_0.
The extent of the entrance (exit) fringe field is characterized by $\lambda_E(\lambda_S)$. The distance of ray-tracing on both sides of the EFBs, in the field fall off regions, will be $\pm\, X_E$ at the entrance, and $\pm\, X_S$ at the exit (Fig. 36), by prior and further automatic change of frame.
In the fringe field regions $[-X_E, X_E]$ and $[-X_S, X_S]$ on both sides of the EFBs, $\vec{B}(X,Y,Z)$ and its derivatives up to fourth order are calculated at each step of the trajectory from the analytical expressions of the three components B_X, B_Y, B_Z obtained by differentiation of the scalar potential (see section 1.3.7) expressed to the 8th order in Y and Z.

$$V(X,Y,Z) = \left(G(X) - \frac{G''(X)}{12}\,(Y^2+Z^2) + \frac{G''''(X)}{384}\,(Y^2+Z^2)^2 - \frac{G''''''(X)}{23040}\,(Y^2+Z^2)^3 \right) YZ$$

$$\left(\quad G^{(n)}(X) = d^n G(X)/dX^n, etc. \right)$$

where $G(X)$ is the gradient on axis [37] :

$$G(X) = \frac{G_0}{1+\exp P(d(X))} \qquad \text{with} \quad G_0 = \frac{B_0}{R_0}$$

and,

$$P(d) = C_0 + C_1\left(\frac{d}{\lambda}\right) + C_2\left(\frac{d}{\lambda}\right)^2 + C_3\left(\frac{d}{\lambda}\right)^3 + C_4\left(\frac{d}{\lambda}\right)^4 + C_5\left(\frac{d}{\lambda}\right)^5$$

where $d(X)$ is the distance to the field boundary and λ stands for λ_E or λ_S (normally, $\lambda \simeq 2 * R_0$).
When fringe fields overlap inside the magnet ($XL \leq X_E + X_S$), the gradient G is expressed as

$$G = G_E + G_S - 1$$

where, G_E is the entrance gradient and G_S is the exit gradient.
If $\lambda_E = 0$ ($\lambda_S = 0$), the field at entrance (exit) is considered as sharp edged, and then $X_E(X_S)$ is forced to zero (for the mere purpose of saving computing time).
Outside of the fringe field regions (or everywhere when $\lambda_E = \lambda_S = 0$) $\vec{B}(X,Y,Z)$ in the magnet is given by

$$\begin{aligned} B_X &= 0 \\ B_Y &= G_0 Z \\ B_Z &= G_0 Y \end{aligned}$$

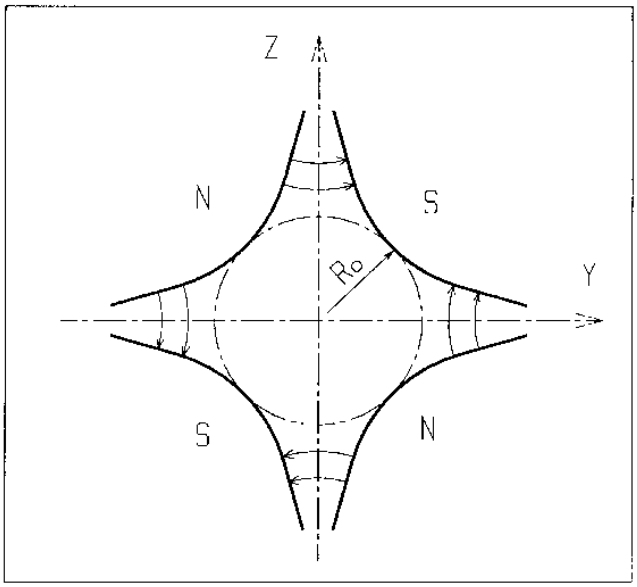


Figure 35: Quadrupole magnet

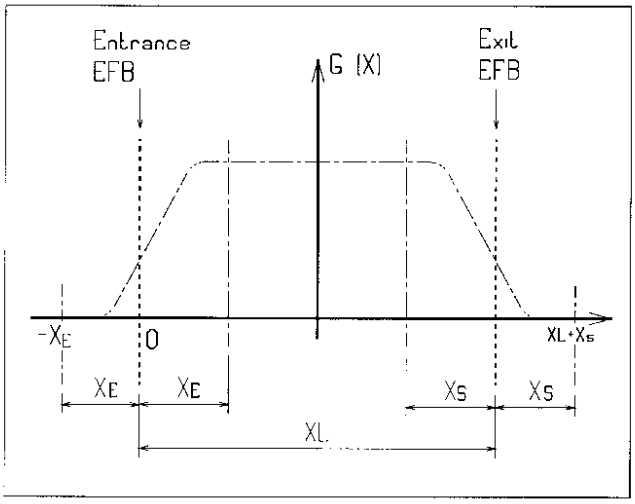


Figure 36: Scheme of the longitudinal field gradient $G(X)$.
(OX) is the longitudinal axis of the reference frame $(0, X, Y, Z)$ of **zgoubi**. The length of the element is *XL*. Trajectories are ray-traced from $-X_E$ to $XL + X_S$, by means of respectively prior and final automatic change of frame.

SEPARA : Wien Filter - analytical simulation

Note : simulation by stepwise integration can be found in *WIENFILTER*.

SEPARA provides an analytic simulation of an electrostatic separator. Input data are the length L of the element, the electric field E and the magnetic field B. The mass m and charge q of the particles are entered by means of the keyword *PARTICUL*.

The subroutines involved in *SEPARA* solve the following system of three equations with three unknown variables S, Y, Z (while $X \equiv L$), that describe the cycloidal motion of a particle in $\vec{E}$, $\vec{B}$ static fields (Fig. 37).

$$X = -R\cos\left(\frac{\omega S}{\beta c} + \epsilon\right) - \frac{\alpha S}{\omega \beta c} + \frac{C_1}{\omega}$$
$$Y = R\sin\left(\frac{\omega S}{\beta c} + \epsilon\right) - \frac{\alpha}{\omega^2} - \frac{C_2}{\omega} + Y_0$$
$$Z = S\sin(P_0) + Z_0$$

where, S is the path length in the separator, $\alpha = -\dfrac{Ec^2}{\gamma}$, $\omega = -\dfrac{Bc^2}{m\gamma}$, $C_1 = \beta\sin(T_0)\cos(P_0)$ and $C_2 = \beta c\cos(T_0)\cos(P_0)$ are initial conditions. c = velocity of light, βc = velocity of the particle, $\gamma = (1-\beta^2)^{-\frac{1}{2}}$ and $\tan\epsilon = (C_2 + \dfrac{\alpha}{\omega})/C_1$. Y_0, T_0, Z_0, P_0 are the initial coordinates of the particle in the **zgoubi** reference frame. Here βc and γ are assumed constant, which is true as long as the change of momentum due to the electric field remains negligible all along the separator.

The option index IA in the input data allows switching to inactive element (thus equivalent to *ESL*), horizontal or vertical separator. Normally, E, B and the value of β_W for wanted particles are related by

$$B(T) = -\frac{E\left(\dfrac{V}{m}\right)}{\beta_W \cdot c\left(\dfrac{m}{s}\right)}$$

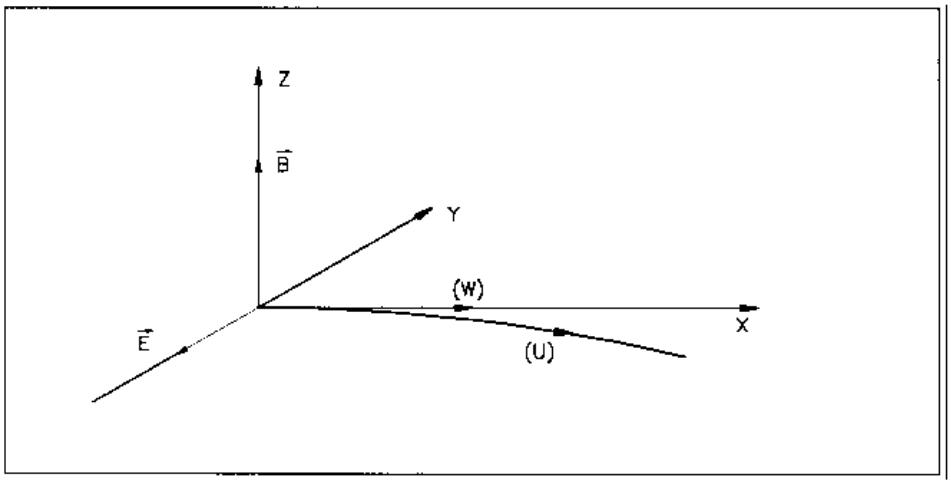


Figure 37: Horizontal separation between a wanted particle, (W), and an unwanted particle, (U). (W) undergoes a linear motion while (U) undergoes a cycloidal motion.

SEXTUPOL : Sextupole magnet (Fig. 38)

The meaning of parameters for *SEXTUPOL* is the same as for *QUADRUPO*.

In fringe field regions the magnetic field $\vec{B}(X,Y,Z)$ and its derivatives up to fourth order are derived from the scalar potential approximated to 7th order in Y and Z

$$V(X,Y,Z) = \left(G(X) - \frac{G''(X)}{16}(Y^2+Z^2) + \frac{G''''(X)}{640}(Y^2+Z^2)^2\right)\left(Y^2Z - \frac{Z^3}{3}\right)$$

$$\text{with } G_0 = \frac{B_0}{R_0^2}$$

The modelling of the fringe field form factor $G(X)$ is described under *QUADRUPO*, p. 147.

Outside fringe field regions, or everywhere in sharp edge sextupole ($\lambda_E = \lambda_S = 0$), $\vec{B}(X,Y,Z)$ in the magnet is given by

$$B_X = 0$$
$$B_Y = 2G_0YZ$$
$$B_Z = G_0(Y^2 - Z^2)$$

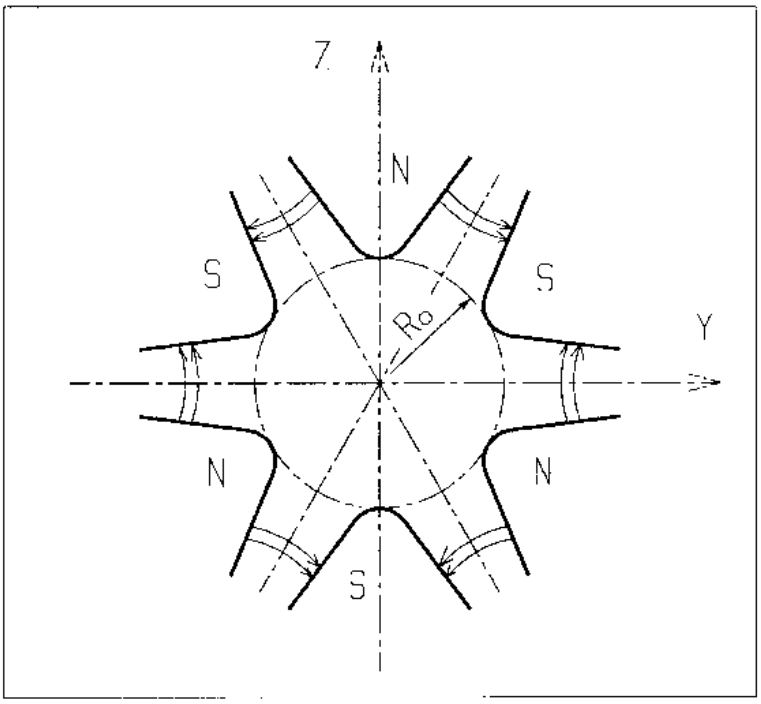


Figure 38: Sextupole magnet

SOLENOID : Solenoid (Fig. 39)

The solenoidal magnet has an effective length *XL*, a mean radius R_0 and an asymptotic field $B_0 = \mu_0 NI/XL$ (*i.e.*, $\int_{-\infty}^{\infty} B_X(X,r)dX = \mu_0 NI,\ \forall r < R_0$), wherein B_X=longitudinal field component, NI = number of ampere-Turns, $\mu_0 = 4\pi 10^{-7}$.

The distance of ray-tracing beyond the effective length *XL*, is X_E at the entrance, and X_S at the exit (Fig. 39).

Two methods are available for the computation of the field $\vec{B}(X,r)$ and its derivatives.

Method 1 : uses the on-axis field model

$$B_X(X, r=0) = \frac{\mu_0 NI}{2XL}\left[\frac{XL/2 - X}{\sqrt{(XL/2-X)^2 + R_0^2}} + \frac{XL/2 + X}{\sqrt{(XL/2+X)^2 + R_0^2}}\right] \tag{5.4.21}$$

with $X = r = 0$ taken at the center of the solenoid. This model assumes that the coil thickness is small compared to the mean radius R_0. In this model, the magnetic length is

$$L_{mag} \equiv \frac{\int_{-\infty}^{\infty} B_X(X, r<R_0)dX}{B_X(X=r=0)} = XL\sqrt{1 + \frac{4R_0^2}{XL^2}} > XL \tag{5.4.22}$$

with in addition

$$B_X(\text{center}) \equiv B_X(X=r=0) = \frac{\mu_0 NI}{XL\sqrt{1 + \frac{4R_0^2}{XL^2}}}.$$

From eq. 5.4.21, the field and its derivatives at all (X,Y,Z) are extrapolated, following the method described in section 1.3.1.

Method 2 : computes $\vec{B}(X,r)$ (with $r = (Y^2+Z^2)^{1/2}$) and its derivatives up to second order at all (X,Y,Z) following the technique in Ref. [54], based on the three complete elliptic integrals K, E and Π. The latter are calculated with the algorithm proposed in the same reference, their derivatives are calculated by means of recursive relations [55].

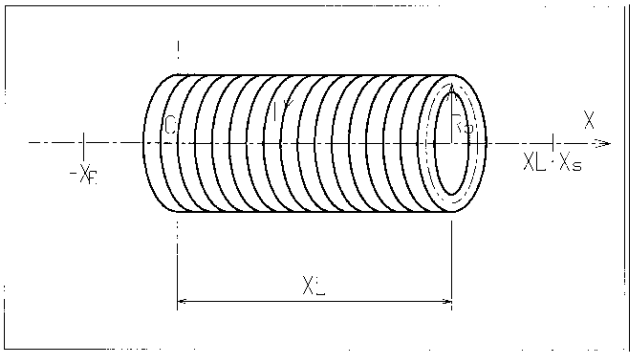


Figure 39: Solenoidal magnet.

For software developers

Solenoid parameters are read from zgoubi.dat by rsolen.f. Initializations prior to tracking done in soleno.f. Field computation during tracking is in solenf.f.

SPINR : Spin rotation [56]

SPINR causes spin precession, a local transformation on the spin vector of the particles, as, *e.g.*, in a local helical spin rotator. Two options are available, as follows.

IOPT = 0 inhibits the keyword.

If IOPT = 1 : the precession is defined by its axis, at angle ϕ with respect to **zgoubi**'s X-axis and assumed in the horizontal plane, and by its value, μ.

If IOPT = 2 : given the spin precession axis direction ϕ in the (X,Y) plane, the spin precession angle follows a function of the *reference* Lorentz factor :

$$\mu(\gamma) = \left(\frac{B}{B_0}\right)^2 \times \left(C_0 + \frac{C_1}{\gamma} + \frac{C_2}{\gamma^2} + \frac{C_3}{\gamma^3}\right)$$

with B and B_0 normally a nominal and a reference magnetic field value, and C_i empirical coefficients. The *reference* Lorentz factor corresponds to the reference rigidity *BORO*, as defined for instance with *[MC]OBJET*, possibly updated in the presence of acceleration (see section 5.6.2, page 179).

TOSCA : 2-D and 3-D Cartesian or cylindrical mesh field map (page 287 in PART B)

TOSCA is dedicated to the reading and treatment of

- 2-D Cartesian or cylindrical mesh mid-plane field maps (mid-plane is assumed anti-symmetry plane, as in an upright multipole),
- 3-D Cartesian or cylindrical mesh field maps.

TOSCA was first installed to read *TOSCA* magnet computer code style of fieldmap output and formatting, hence the name. However the actual origin of the fieldmap, TOSCA or other computer code, does not matter.

TOSCA has been developed over the years to allow reading many different input formats, including 3-D maps given under the form of a series of files, partial 3-D maps with particular symmetry plane(s), etc. In order to select between the many possibilities, two input data, IZ and MOD.MOD2, are used. This is addressed in the following, and table 1, page 154, lists all possibilities as a function of the values given to IZ and MOD.MOD2.

A double-flag, *MOD[.MOD2]*, determines whether Cartesian or Z-axis cylindrical mesh is used, and the nature and/or form of the field map data set.

The total number of field data files to be read is determined by the *MOD[.MOD2]* flag (see below) and by the parameter IZ that appears in the data list following the keyword. Each of these files contains the field components B_X, B_Y, B_Z on an (X, Y) mesh. $IZ = 1$ for a 2-D map, and in this case B_X and B_Y are assumed zero all over the map[8]. For a 3-D map with mid-plane symmetry, described with a set of 2-D maps at various Z, then *MOD=0* and $IZ \geq 2$, and thus, the first data file whose name follows in the data list is supposed to contain the median plane field (assuming $Z = 0$ and $B_X = B_Y = 0$), while the remaining $IZ - 1$ file(s) contain the $IZ - 1$ additional planes in increasing Z order. For arbitrary 3-D maps, no symmetry assumed, then *MOD=1* and the total number of maps (whose names follow in the data list) is IZ, such that map number $[IZ/2] + 1$ is the $Z = 0$ elevation one.

The field map data file has to be be filled with a format that fits the *FORTRAN* reading sequence. The following is an instance, details and possible updates are to be found in the source file `'fmapw.f'` :

```
DO  1  K = 1, KZ
  OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
  DO  1  J = 1,  JY
    DO  1 I = 1, IX
              IF (BINARY) THEN
                READ(NL) Y(J), Z(K), X(I), BY(J,K,I), BZ(J,K,I), BX(J,K,I)
              ELSE
        READ(NL,100) Y(J), Z(K), X(I), BY(J,K,I), BZ(J,K,I), BX(J,K,I)
 100            FORMAT(1X,6E11.2)
              ENDIF
  1     CONTINUE
```

IX (JY, KZ) is the number of longitudinal (transverse horizontal, vertical) nodes of the 3-D uniform mesh. For letting **zgoubi** know in case these are binary files, FNAME must begin with 'B_' or 'b_'.

Other *MOD[.MOD2]* cases in addition to the above, are listed in table 1. The *FORTRAN* subroutine `fmapw.f` and its entries FMAPR, FMAPR2 can be looked up for more details, in particular regarding the formatting of the field map data file(s).

Once the field map(s) reading is completed, the field $\vec{B} = (B_X, B_Y, B_Z)$ is normalized by means of *BNORM* in a similar way as in *CARTEMES*. As well the coordinates X and Y (and Z in the case of 3-D field maps) are normalized by the *X-[, Y-, Z-]NORM* coefficient (useful to convert to centimeters, the working units in **zgoubi**).

At each step of the trajectory of a particle inside the map, the field and its derivatives are calculated as follows :

[8]Use *MAP2D* in case non-zero B_X, B_Y are to be taken into account in a 2-D map.

Table 1: The various *IZ, MOD* and *MOD2* possibilities, when using *TOSCA*.

IZ : number of nodes of the *complete* field map along the Z direction (IZ=1 for 2D)
MOD, MOD2 : determine the coordinate system, symmetries, reading format and column sequence, etc.
NF : number of field map input data files to be declared. Always include mid-plane map.
Expected columns : formatting of the coordinates and field data columns in the field map data file(s)
'Exemple' example folder : examples of **zgoubi** runs using field maps can be found in the subfolders of zgoubi-code/exemples/KEYWORDS/TOSCA/cartesian (case MOD$\leq$19) or zgoubi-code/exemples/KEYWORDS/TOSCA/cylindrical (case MOD$\geq$20). The rightmost column below indicates the subfolder of concern, following *(IZ, MOD, MOD2)* options.

MOD $\leq$ 19 : Cartesian mesh

IZ	MOD	.MOD2		NF	Expected columns	Example folder
1	0, 1	none or .1, .2, .3	**2-D** map. File contains $B_Z(X,Y)\|_{Z=0}$, mid-plane antisymmetry assumed. Several different reading formats (see fmapw.f/fmapr3).	1	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_1-0-none/ (GSI KAOS spectrometer)
1	3	none *or* .1	**2-D** map. Used for AGS main magnet. If MOD2=1, $B_Z(X,Y,Z=0)$ field is perturbed by $(1+n_1\,Y+n_2\,Y^2+n_3\,Y^3)$ factor.	1	Special - see example	AGS/usingMainMagnetsMaps (AGS with main magnet maps)
1	15	$.1-.4$	**2-D** map. Up to 4 files to be combined linearly into a new map : field at all node of new map is $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Mid-plane antisymmetry is assumed : each file has to contain $B_Z(X,Y,Z=0)$.	$1-4$	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_1-15-.1-.4 (EMMA FFAG cell)
>1	0	none	**3-D** map. Files span upper half of magnet, one per (X,Y)$\|_{0\leq Z\leq Z_{max}}$ plane including median plane, mid-plane antisymmetry assumed.	1+IZ/2	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_gt1-0-none/ (GSI KAOS spectrometer)
2p+1 $p\geq 1$	1	none	**3-D** map. Files span full magnet volume, one file per (X,Y) plane, no symmetry assumed.	IZ	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_gt1-1-none/ (AGS warm helix snake)
>1	12	none	**3-D** map. Single file, upper half of magnet, mid-plane antisymmetry assumed.	1		
>1	12	.1	**3-D** map. Single file, whole magnet volume, no symmetry assumed.	1	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_gt1-12-.1/ (AGS warm helix snake)
>1	12	.2	**3-D** map. Single file, 1/8th of the magnet, symmetry wrt. $(X,Y)_{Z=0}$, $(X,Z)_{Y=0}$, $(Y,Z)_{X=0}$ planes.	1		
2p+1 $p\geq 1$	15	$.1-.4$	**3-D** map. Up to 4 files to be combined linearly into a new map, field at all node of new map is $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file has to contain $B_{X,Y,Z}(X,Y,Z)$ data over *IZ* equally Z-spaced (X,Y) planes (no symmetry assumed).	$1-4$		
2p+1 $p\geq 1$	16	$.1-.4$	**3-D** map. Fields from up to 4 maps to be combined linearly into a new field value at particle location, $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file has to contain $B_{X,Y,Z}(X,Y,Z)$ data over *IZ* equally Z-spaced (X,Y) planes.	$1-4$	UNDER DEVELOPMENT	

MOD $\geq$ 20 : Cylindrical mesh

IZ	MOD	.MOD2		NF	Expected columns	Example folder
1	25	$.1-.4$	**2-D** map. Up to 4 files to be combined linearly into a new map : at all node of new map $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file contains mid-plane $B_Z(X,Y,Z=0)$ data, mid-plane antisymmetry is assumed.	$1-4$	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_1-15-.1-.4 (EMMA FFAG cell)
>1	20, 21		**3-D** map. Single file. MOD=20 : 1/4 magnet, cylindrical symmetry with respect to (Y,Z) plane and antisymmetry wrt (X,Y) plane. MOD=21 : another type of symmetry (to be documented - see fmapw.f).	1	Y(r,θ), Z, X(r,θ), BY, BZ, BX	IZ-MOD-.MOD2_gt1-20 (KEK 150 MeV FFAG)
2p+1 $p\geq 0$	22, 23	$.1-.4$	**2D** or **3-D** map. Mid-plane antisymmetry assumed. Up to 4 files can be combined linearly into a new one, $Z\geq 0$ half-magnet volume each ; field at all node of new map is $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file has to contain $B_{X,Y,Z}(X,Y,Z)$ data over *IZ* equally Z-spaced (X,Y) planes. MOD=23 : special, test code (see fmapw.f).	$1-4$	Y(r,θ), Z, X(r,θ), BY, BZ, BX	
2p+1 $p\geq 1$	24		**3-D** map, full volume. No symmetry assumed.	1	θ, R, Z, B_θ, BR, BZ	

- in the case of a 2-D map, by means of a second or fourth order polynomial interpolation, depending on *IORDRE* (*IORDRE* = 2, 25 or 4), as for *CARTEMES*,

- in the case of a 3-D map, by means of a second order polynomial interpolation with a $3 \times 3 \times 3$-point parallelepipedic grid, as described in section 1.4.4.

In the Cartesian mesh case, entrance and/or exit integration boundaries between which the trajectories are integrated in the field may be defined, in the same way as in *CARTEMES*.

A *'TITL'* (a line of comments) is part of the arguments of the keyword *TOSCA*. It allows introducing options, for instance :

◇ including 'HEADER n' allows specifying the number of header lines ('n' non-data lines) at the top of the field map file,

◇ including 'FLIP' in *TITL* causes the field map to be X-flipped,

◇ including 'ZroBXY' forces $B_X = B_Y = 0$ at all Z=0 nodes of the field map mesh (only applies with *MOD*=15 and *MOD*=24),

◇ including 'RHIC_helix' will normalize *BNORM* to measured B field value versus helix current (measured B(I) data are hard-coded in program toscac.f). In that case, *BNORM* may be given the current value, in Amps, while a(1) in mode *MOD.MOD2=15.1* is used to normalize the field map (namely, $a(1) = 1/B_{max}$ with B_{max} the maximum field value on helix axis).

TRANSMAT : Matrix transfer

TRANSMAT performs a second order transport of the particle coordinates in the following way

$$X_i = \sum_j R_{ij} X_j^0 + \sum_{j,k} T_{ijk} X_j^0 X_k^0$$

where, X_i stands for any of the current coordinates Y, T, Z, P, path length and momentum dispersion, and X_i^0 stands for any of the initial coordinates. $[R_{ij}]$ ($[T_{ijk}]$) is the first order (second order) transfer matrix as usually involved in second order beam optics [31]. Second order transfer is optional. The length of the element represented by the matrix may be introduced for the purpose of path length updating.

Note : *MATRIX* delivers $[R_{ij}]$ and $[T_{ijk}]$ matrices in a format suitable for straightforward use with *TRANSMAT*.

TRAROT : Translation-Rotation of the reference frame

UNDER DEVELOPMENT. Check before use.

This procedure performs translation and rotation of the local **zgoubi** frame. It can be used for instance for skewing multipoles.

Relationship to spin tracking, particle decay or gas-scattering may not be fully installed, to be checked before use.

UNDULATOR : Undulator magnet

UNDULATOR magnet. UNDER DEVELOPMENT.

UNIPOT : Unipotential cylindrical electrostatic lens

The lens is cylindrically symmetric about the X-axis.

The length of the first (resp. second, third) electrode is $X1$ (resp. $X2$, $X3$). The distance between the electrodes is D. The potentials are $V1$ and $V2$. The inner radius is R_0 (Fig. 40). The model for the electrostatic potential along the axis is [57]

$$V(x) = \frac{V2 - V1}{2\omega D} \left[\ln \frac{\cosh \dfrac{\omega \left(x + \dfrac{X2}{2} + D \right)}{R_0}}{\cosh \dfrac{\omega \left(x + \dfrac{X2}{2} \right)}{R_0}} + \ln \frac{\cosh \dfrac{\omega \left(x - \dfrac{X2}{2} - D \right)}{R_0}}{\cosh \dfrac{\omega \left(x - \dfrac{X2}{2} \right)}{R_0}} \right]$$

($x =$ distance from the center of the central electrode ; ω = 1,318 ; cosh = hyperbolic cosine), from which the field $\vec{E}(X, Y, Z)$ and its derivatives are deduced following the procedure described in section 1.3.1.

Use *PARTICUL* prior to *UNIPOT*, for the definition of particle mass and charge.

The total length of the lens is $X1 + X2 + X3 + 2\,D$; stepwise integration starts at entrance of the first electrode and terminates at exit of the third one.

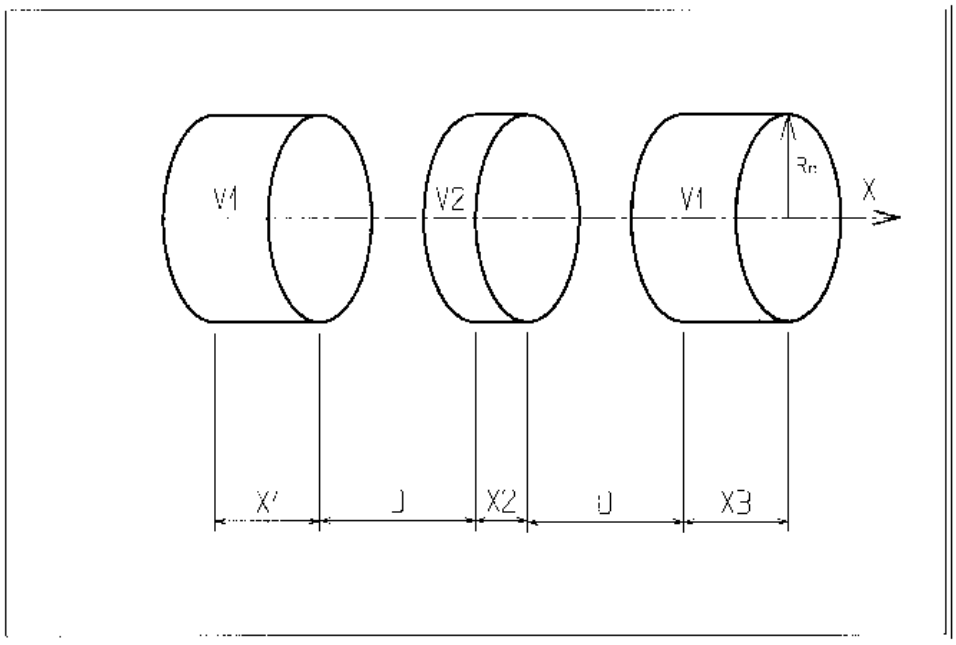


Figure 40: Three-electrode cylindrical unipotential lens.

VENUS : Simulation of a rectangular shape dipole magnet

VENUS is dedicated to a 'rough' simulation of SATURNE Laboratory's *VENUS* dipole. The field B_0 is constant inside the magnet, with longitudinal extent *XL* and transverse extent $\pm YL$; outside these limits, $B_0 = 0$ (Fig. 41).

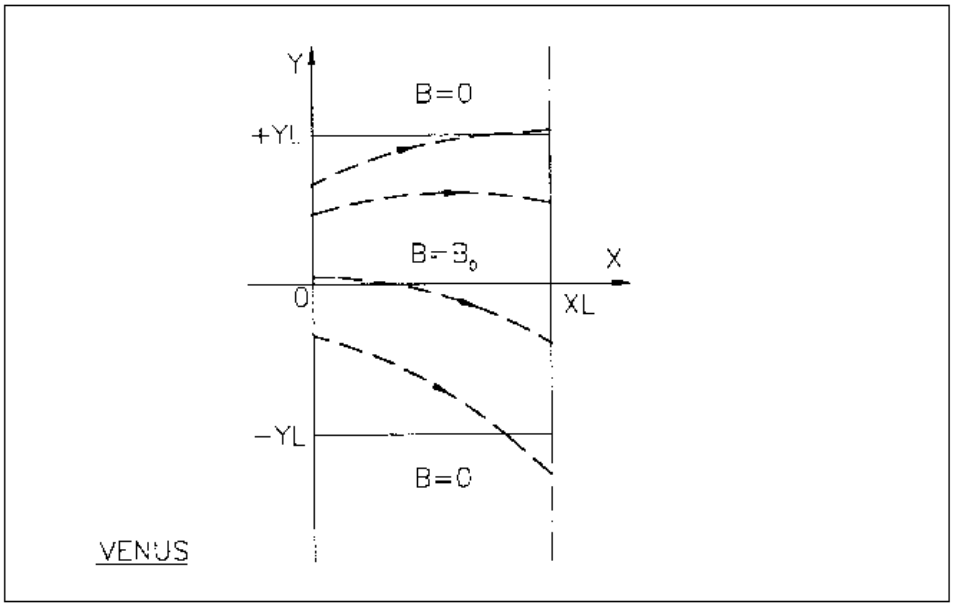


Figure 41: Scheme of *VENUS* rectangular dipole.

WIENFILT : Wien filter

WIENFILT simulates a Wien filter, with transverse and orthogonal electric and magnetic fields $\vec{E}_Y$, $\vec{B}_Z$ or $\vec{E}_Z$, $\vec{B}_Y$ (Fig. 37). It must be preceded by *PARTICUL* for the definition of particle mass and charge.

The length *XL* of the element is the distance between its entrance and exit EFBs. The electric and magnetic field intensities E_0 and B_0 in the central, uniform field region, normally satisfy the relation

$$B_0 = -\frac{E_0}{\beta_W c}$$

for the selection of " wanted" particles of velocity $\beta_W c$. Ray-tracing in field fall-off regions extends over a distance X_E (X_S) beyond the entrance (exit) EFB by means of prior and further automatic change of frame. Four sets of coefficients λ, $C_0 - C_5$ allow the description of the entrance and exit fringe fields outside the uniform field region, following the model [37]

$$F = \frac{1}{1 + \exp(P(s))}$$

where $P(s)$ is of the term

$$P(s) = C_0 + C_1 \left(\frac{s}{\lambda}\right) + C_2 \left(\frac{s}{\lambda}\right)^2 + C_3 \left(\frac{s}{\lambda}\right)^3 + C_4 \left(\frac{s}{\lambda}\right)^4 + C_5 \left(\frac{s}{\lambda}\right)^5$$

and s is the distance to the EFB. When fringe fields overlap inside the element (*i.e.*, *XL* $\leq X_E + X_S$), the field fall-off is expressed as

$$F = F_E + F_S - 1$$

where $F_E(F_S)$ is the value of the coefficient respective to the entrance (exit) EFB.
If $\lambda_E = 0$ ($\lambda_S = 0$) for either the electric or magnetic component, then both are considered as sharp edge fields and $X_E(X_S)$ is forced to zero (for the purpose of saving computing time). In this case, the magnetic wedge angle vertical first order focusing effect is simulated at entrance and exit by a kick $P_2 = P_1 - Z_1 \tan(\epsilon/\rho)$ applied to each particle (P_1, P_2 are the vertical angles upstream and downstream the EFB, Z_1 the vertical particle position at the EFB, ρ the local horizontal bending radius and ϵ the wedge angle experienced by the particle ; ϵ depends on the horizontal angle T). This is not done for the electric field, however it is advised not to use a sharp edge electric dipole model since this entails non symplectic mapping, and in particular precludes accounting for momentum effects of the non zero longitudinal electric field component.

YMY : Reverse signs of Y and Z reference axes

YMY performs a 180° rotation of particle coordinates with respect to the X-axis, as shown in Fig. 42. This is done by means of a change of sign of Y and Z axes, and therefore coordinates, as follows

$$Y2 = -Y1, \quad T2 = -T1, \quad Z2 = -Z1 \quad \text{and} \quad P2 = -P1$$

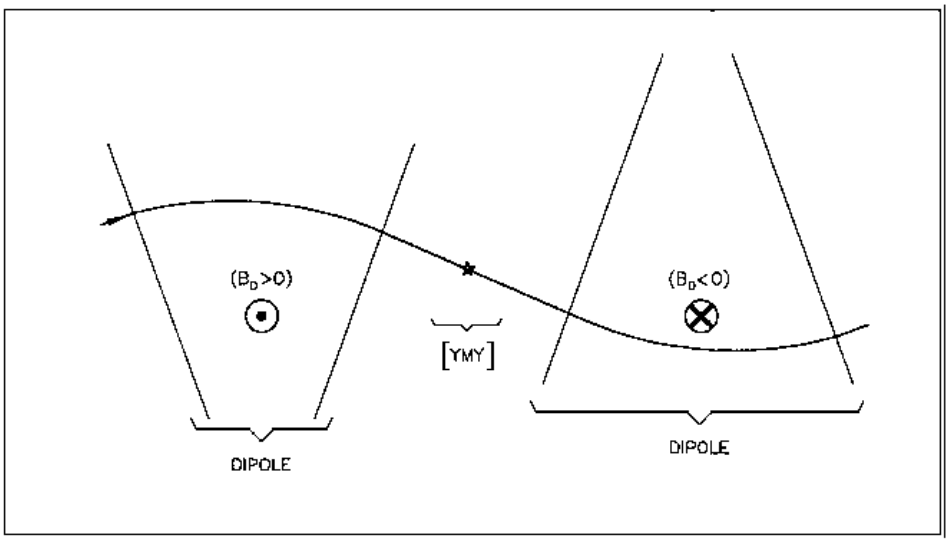


Figure 42: The use of YMY in a sequence of two identical dipoles of opposite signs.

5.5 Output Procedures

These procedures are dedicated to the storage or printing of particle coordinates, histograms, spin coordinates, etc. They may be called for at any spot in the data pile.

5.5.1 FAISCEAU, FAISCNL, FAISTORE : Print/Store particle coordinates

• *FAISCEAU* can be introduced anywhere in a structure data list (zgoubi.dat). It produces a print (into zgoubi.res) of initial and actual coordinates of the *IMAX* particles at the location where it stands, together tagging indices and letters, etc.

• *FAISCNL* produces a lot more information on particles at current location, including spin components, decay distance, mass, charge, etc. (see list below), and stores it in a dedicated file *FNAME* (advised name is *FNAME* = 'zgoubi.fai' (formatted write) or 'b_zgoubi.fai' (binary write) if post-processing with **zpop** should follow). This file may further on be read by means of *OBJET*, option *KOBJ*= 3, or used for other purposes such as graphics (see Part D of the Guide).
The data written to that file are formatted and ordered according to the *FORTRAN* sequence in the subroutine `impfai.f`, where details and possible updates are to found. The following is an instance :

```
    OPEN (UNIT = NL, FILE = FNAME, STATUS = 'NEW')
      IF(BINARY) THEN
        DO 2 I=1,IMAX
            P = BORO*CL9 *F(1,I) * AMQ(2,I)
            ENERG = SQRT(P*P + AMQ(1,I)*AMQ(1,I))
            ENEKI = ENERG - AMQ(1,I)
            WRITE(NFAI)
     1      IEX(I),-1.D0+FO(1,I),(FO(J,I),J=2,MXJ),                 tag, initial D-1,Y,T,Z,P,S,Time
     2      -1.D0+F(1,I),F(2,I),F(3,I),(F(J,I),J=4,MXJ),            current D_i,Y,T,Z,P,S,Time
     >      (SI(J,I),J=1,4),(SF(J,I),J=1,4),                        spin components, initial and current
     >      ENEKI,ENERG,                                            energy
     4      I,IREP(I), SORT(I),(AMQ(J,I),J=1,5),RET(I),DPR(I),PS,   particle#,loss S,mass,Q,G,life time
     5      BORO, IPASS, NOEL, KLEY,LBL1,LBL2,LET(I)                BORO, pass#,element#,keyword,labels
 2      CONTINUE
       ELSE
        DO 1 I=1,IMAX
          P = BORO*CL9 *F(1,I) * AMQ(2,I)
          ENERG = SQRT(P*P + AMQ(1,I)*AMQ(1,I))
          ENEKI = ENERG - AMQ(1,I)
          WRITE(NFAI,110)
     1    IEX(I),-1.D0+FO(1,I),(FO(J,I),J=2,MXJ),
     2    -1.D0+F(1,I),F(2,I),F(3,I),
     3    (F(J,I),J=4,MXJ),
     4    (SI(J,I),J=1,4),(SF(J,I),J=1,4),
     5    ENEKI,ENERG,
     6    I,IREP(I), SORT(I),(AMQ(J,I),J=1,5),RET(I),DPR(I),PS,
     7    BORO, IPASS, NOEL,
     8    TX1,KLEY,TX1,TX1,LBL1,TX1,TX1,LBL2,TX1,TX1,LET(I),TX1
          INCLUDE "FRMFAI.H"
 1      CONTINUE
       ENDIF

 110     FORMAT(1X,

C1               KEX,      XXXO,(FO(J,IT),J=2,MXJ)
     >  1P,      1X,I2,    7(1X,E16.8)

C2              XXX,Y,T*1.D3,
     >         ,3(1X,E24.16)

C3       Z,P*1.D3,SAR,TAR
     >   ,4(1X,E24.16)

C4       SXo, SYo, SZo, So, SX, SY, SZ, S
     >   ,8(1X,E15.7)

C5        ENEKI,ENERG
     >   ,2(1X,E16.8)

C6       IT,IREP(IT),   SORT(IT), (AMQ(J,I),J=1,5), RET(IT), DPR(IT), PS
     >   ,2(1X,I6),   9(1X,E16.8)

C7         BORO,   IPASS, NOEL,
     >   ,1X,E16.8,  2(1X,I6)

C8            'KLEY',    ('LABEL(NOEL,I)',I=1,2),           'LET(IT)'
     >    ,1X,A1,A10,A1,   2(1X,A1,A ,A1),                   1X,3A1)
```

The meaning of the main data is the following (see the keyword *OBJET*)

$LET(I)$	:	one-character string, for tagging particle number I
IEX, I, $IREP(I)$	:	flag, particle number, index
$FO(1-6,I)$	:	coordinates D, Y, T, Z, P and path length at the origin of the structure
$F(1-6,I)$	:	*idem* at the current position
$SORT(I)$	:	path length at which the particle has possibly been stopped (see *CHAMBR* or *COLLIMA*)
$RET(I)$, $DPR(I)$	:	synchrotron phase space coordinates ; *RET* =phase (radian), *DPR* = momentum dispersion (MeV/c) (see *CAVITE*)
IPASS	:	turn number (see *REBELOTE*)
etc.	:	

• *FAISTORE* has an effect similar to *FAISCNL*, with two more features.

- On the first data line, *FNAME* may be followed by a series of up to 10 *LABEL*s. If there is no label, the print occurs by default at the location of *FAISTORE* ; if there are labels the print occurs right downstream of all optical elements wearing those labels (and no longer at the *FAISTORE* location).

- The next data line gives a parameter, IP : printing will occur at pass 1 and then at every IP other pass, if using *REBELOTE* with *NPASS* $\geq IP-1$.

For instance the following data input in zgoubi.dat :

```
FAISTORE
zgoubi.fai      HPCKUP    VPCKUP
12
```

will result in output prints into zgoubi.fai, at pass 1 and then at every 12 other pass, each time elements of the zgoubi.dat data list labeled either *HPCKUP* or *VPCKUP* are encountered.

Note

Binary storage can be obtained from *FAISCNL* and *FAISTORE*. This is for the sake of compactness and access speed, for instance in case voluminous amounts of data would have to be manipulated using **zpop**.

This is achieved by giving the storage file a name of the form *b_FNAME* or *B_FNAME* (*e.g.*, 'b_zgoubi.fai'). The *FORTRAN WRITE* list is the same as in the *FORMATTED* case above.

This is compatible with the *READ* statements in **zpop** that will recognize binary storage from that very radical 'b_' or 'B_'.

5.5.2 FOCALE, IMAGE[S] : Particle coordinates and beam size ; localization and size of horizontal waist

FOCALE calculates the dimensions of the beam and its mean transverse position, at a longitudinal distance *XL* from the position corresponding to the keyword *FOCALE.*

IMAGE computes the location and size of the closest horizontal waist.

IMAGES has the same effect as *IMAGE*, but, in addition, for a non-monochromatic beam it calculates as many waists as there are distinct momenta in the beam, provided that the object has been defined with a classification of momenta (see *OBJET*, *KOBJ*= 1, 2 for instance).

Optionally, for each of these three procedures, **zgoubi** can list a trace of the coordinates in the X, Y and in the Y, Z planes.

The following quantities are calculated for the N particles of the beam (*IMAGE, FOCALE*) or of each group of momenta (*IMAGES*)

- Longitudinal position :

$$\begin{aligned} \textit{FOCALE}: \quad & X = \textit{XL} \\ \textit{IMAGE[S]}: \quad & X = -\frac{\sum_{i=1}^{N} Y_i * \mathrm{tg}T_i - \left(\sum_{i=1}^{N} Y_i * \sum_{i=1}^{N} \mathrm{tg}T_i\right) /N}{\sum_{i=1}^{N} \mathrm{tg}^2 T_i - \left(\sum_{i=1}^{N} \mathrm{tg}T_i\right)^2 /N} \\ & Y = Y_1 + X * \mathrm{tg}T_1 \end{aligned}$$

  where Y_1 and T_1 are the coordinates of the first particle of the beam (*IMAGE, FOCALE*) or the first particle of each group of momenta (*IMAGES*).

- Transverse position of the center of mass of the waist (*IMAGE[S]*) or of the beam (*FOCALE*), with respect to the reference trajectory

$$\textit{YM} = \frac{1}{N}\sum_{i=1}^{N}(Y_i + X\mathrm{tg}T_i) - Y = \frac{1}{N}\sum_{i=1}^{N} Y\ M_i$$

- FWHM of the image (*IMAGE[S]*) or of the beam (*FOCALE*), and total width, respectively, W and WT

$$\begin{aligned} W &= 2.35\left(\frac{1}{N}\sum_{i=1}^{N} Y\ M_i^2 - Y\ M^2\right)^{\frac{1}{2}} \\ WT &= \max(\textit{YM}_i) - \min(\textit{YM}_i) \end{aligned}$$

5.5.3 FOCALEZ, IMAGE[S]Z : Particle coordinates and beam size ; localization and size of vertical waist

Similar to *FOCALE* and *IMAGE[S]*, but the calculations are performed with respect to the vertical coordinates Z_i and P_i, in place of Y_i and T_i.

5.5.4 HISTO : 1-D histogram

Any of the coordinates used in **zgoubi** may be histogrammed, namely initial Y_0, T_0, Z_0, P_0, S_0, D_0 or current Y, T, Z, P, S, D particle coordinates ($S =$ path length ; D may change in decay process simulation with *MCDESINT*, or when ray-tracing in $\vec{E}$ fields), and also spin coordinates and modulus S_X, S_Y, S_Z and $\left\|\vec{S}\right\|$.

HISTO can be used in conjunction with *MCDESINT*, for statistics on the decay process, by means of *TYP*. *TYP* is a one-character string. If it is set equal to 'S', only secondary particles (they are tagged with an 'S') will be histogrammed. If it is set equal to 'P', then only parent particles (non-'S') will be histogrammed. For no discrimination between S-econdary and P-arent particles, *TYP* = 'Q' must be used.

The dimensions of the histogram (number of lines and columns) may be modified. It can be normalized with *NORM* = 1, to avoid saturation.

Histograms are indexed with the parameter *NH*. This allows making independent histograms of the same coordinate at several locations in a structure. This is also useful when piling up problems in a single input data file (see also *RESET*). *NH* is in the range 1-5.

If *REBELOTE* is used, the statistics on the $1+$*NPASS* runs in the structure will add up.

5.5.5 IMAGE[S][Z] : Localization and size of vertical waists

See FOCALE[Z].

5.5.6 MATRIX : Calculation of transfer coefficients, periodic parameters

MATRIX causes the calculation of the transfer coefficients through the optical structure, from the *OBJET* down to the location where *MATRIX* is introduced in the structure, or, upon option, down to the horizontal focus closest to that location. In this last case the position of the focus is calculated automatically in the same way as the position of the waist in *IMAGE*. Depending on *OBJET* and on option *IFOC*, *MATRIX* also delivers the beam matrix and betatron phase advances or (case of a periodic structure) periodic beam matrix and tunes, chromaticities and other global parameters.

Depending on the value of option *IORD*, different procedures follow

- If *IORD* = 0, *MATRIX* is inhibited (equivalent to *FAISCEAU*, whatever *IFOC*).
- If *IORD* = 1, using *OBJET*, *KOBJ* = 5[.N], the first order transfer matrix $[R_{ij}]$ is calculated, from a third order approximation of the coordinates. For instance

$$Y^+ = \left(\frac{Y}{T_0}\right) T_0 + \left(\frac{Y}{T_0^2}\right) T_0^2 + \left(\frac{Y}{T_0^3}\right) T_0^3, \qquad Y^- = -\left(\frac{Y}{T_0}\right) T_0 + \left(\frac{Y}{T_0^2}\right) T_0^2 - \left(\frac{Y}{T_0^3}\right) T_0^3$$

  will yield, neglecting third order terms,

$$R_{11} = \left(\frac{Y}{T_0}\right) = \frac{Y^+ - Y^-}{2T_0}$$

  This is repeated N times if *OBJET*, *KOBJ* = 5.N ($2 \leq N \leq 99$) is used, so delivering first order data for each 11-particle set, with for each set the reference trajectory being trajectory number 1+(N-1)×11 in the N×11-particle list (keyword *FAISCEAU* will print out that list in zgoubi.res, if desired).

  If *OBJET*, *KOBJ* = 5.1 is used instead (hence introducing initial optical function values, $\alpha_{Y,Z}$, $\alpha_{Y,Z}$, $D_{Y,Z}$, $D'_{Y,Z}$), then, using the R_{ij} above, *MATRIX* will transport the optical functions and phase advances ϕ_Y, ϕ_Z, following

$$\begin{pmatrix} \beta \\ \alpha \\ \gamma \end{pmatrix}_{at\ MATRIX} = \begin{pmatrix} R_{11}^2 & -2R_{11}R_{12} & R_{12}^2 \\ -R_{11}R_{21} & R_{12}R_{21} & R_{11}R_{12} \\ R_{21}^2 & -2R_{21}R_{22} & R_{22}^2 \end{pmatrix} \begin{pmatrix} \beta \\ \alpha \\ \gamma \end{pmatrix}_{at\ OBJET}$$

$$\Delta\phi_Y = A\tan \frac{R_{12}}{(R_{11}\beta_{Y,objet} - R_{12}\alpha_{Y,objet})}, \quad \Delta\phi_Z = A\tan \frac{R_{34}}{(R_{33}\beta_{Z,objet} - R_{34}\alpha_{Z,objet})}, \qquad (5.5.1)$$

$$\phi_{Y,Z} \rightarrow \phi_{Y,Z} + 2\pi \quad \text{if } \phi_{Y,Z} < 0, \text{given } [0, \pi] \text{ Atan determination}$$

  and print these out.

- If *IORD* = 2, using *OBJET*, *KOBJ* = 6, fifth order Taylor expansions are used for the calculation of the first order transfer matrix $[R_{ij}]$ and of the second order matrix $[T_{ijk}]$. Other higher order coefficients are also calculated.
- If *IORD* = 3, using *OBJET*, *KOBJ* = 6.1, transport coefficients up to 3rd order are computed using 102 rays (after routines developed for RAYTRACE [58, 59]).

An automatic generation of an appropriate object for the use of *MATRIX* can be obtained using the procedure *OBJET* (pages 53, 253), as follows
- if *IORD* = 1, use *OBJET*(KOBJ = 5[.N, $2 \leq N \leq 99$]), that generates up to 99*11 initial coordinates. In this case, up to 99 matrices may be calculated, each one *wrt.* to the reference trajectory of concern (trajectory number 1, 12, 23, ... 1+(N-1)×11 respectively).

- if *IORD* = 2, use *OBJET*(KOBJ = 6) that generates 61 initial coordinates.
- if *IORD* = 3, use *OBJET*(KOBJ = 6.1) that generates 102 initial coordinates.

The next option, *IFOC*, acts as follows

- If *IFOC* = 0, the transfer coefficients are calculated at the location of *MATRIX*, and with respect to the reference trajectory. For instance, Y^+ and T^+ above are defined for particle number i as $Y^+ = Y^+(i) - Y(Ref)$, and $T^+ = T^+(i) - T(\text{ref.})$.
- If *IFOC* = 1, the transfer coefficients are calculated at the horizontal focus closest to *MATRIX* (determined automatically), while the reference direction is that of the reference particle. For instance, Y^+ is defined for particle number i as $Y^+ = Y^+(i) - Y_{\text{focus}}$, while T^+ is defined as $T^+ = T^+(i) - T(\text{ref.})$).
- If *IFOC* = 2, no change of reference frame is performed : the coordinates refer to the current frame. Namely, $Y^+ = Y^+(i)$, $T^+ = T^+(i)$, etc.

Periodic Structures

- If *IFOC* = 10 + *NPeriod*, then, from the 1-turn transport matrix as obtained in the way described above, *MATRIX* calculates periodic parameters characteristic of the structure such as optical functions and tune numbers, assuming that it is *NPeriod*-periodic, and in the coupled hypothesis, based on the Edwards-Teng method [33]. This only makes senss when met *at the end* of the zgoubi.dat sequence.

  This is repeated N times if *OBJET, KOBJ* = 5.N ($2 \leq N \leq 99$) is used, so delivering first order data for each 11-particle set.

  If *IORD* = 2 (using *OBJET, KOBJ* = 6) additional periodic parameters are computed such as chromaticities, beta-function momentum dependence, etc.

PRINT : Addition of "*PRINT*" following *IORD, IFOC [, coupled]* will cause stacking of *MATRIX* output data into zgoubi.MATRIX.out file (convenient for use with *e.g.* gnuplot type of data treatment software).

Addition of "*coupled*" next to *IORD, IFOC [, PRINT]*, in the case of periodic beam matrix request (i.e., *IFOC* = 10 + *NPeriod*) will cause use of coupled formalism.

About the source code :

The program `matric` computes the transport matrix coefficients, it is called by `zgoubi` when the keyword *MATRIX* is met along the zgoubi.dat sequence. `matimp`, called by `matric`, ensures the print out of the matrix (and possibly the beam matrix) in zgoubi.res, and upon *'PRINT'* option in zgoubi.MATRIX.out as well.

5.5.7 PICKUPS : Bunch centroid path; orbit

PICKUPS computes the average values of the coordinates of the particles in a bunch (coordinates of the bunch centroid) at one or more keywords specified by their *LABEL*s . The list of *LABEL*(s) concerned is specified by the user, as part of the arguments under the keyword *PICKUPS*.

In conjunction with *REBELOTE* in the case of a periodic structure, with *REBELOTE* thus being used to cause multi-turn tracking, *PICKUPS* are zeroed at start of each turn.

Pickup data computed during **zgoubi** execution are stored in the file zgoubi.PICKUP.out, usable for further analysis, or for plotting (for instance, using gnuplot).

A summary of the list of pickups and their individual bunch data is listed in the result file zgoubi.res (below the *END* keyword) at the end of **zgoubi** execution. It is concluded with statistics as the *rms*, minimum and maximum values of the bunch centroid data over the ensemble of pickups.

5.5.8 PLOTDATA : Intermediate output for the PLOTDATA graphic software

PLOTDATA was at the origin implemented for the purpose of plotting particle coordinates using the TRIUMF PLOTDATA package [60]. However nothing precludes using it with a different aim.

The *PLOTDATA* keyword can be introduced at up to 20 locations in zgoubi.dat. There, particle coordinates will be stored in a local array, `FF`. They are overwritten at each pass. Usage of `FF` is left to the user, see *FORTRAN* subroutine `pltdat.f`.

5.5.9 SPNPRNL, SPNSTORE : Print/Store spin coordinates

• *SPNPRNL* has similar effect to *SPNPRT* (page 175), except that the information is stored in a dedicated file *FNAME* (should post-processing with **zpop** follow, advised name is *FNAME* = 'zgoubi.spn' (formatted write) or 'b_zgoubi.spn' (binary write)). The data are formatted and ordered according to the *FORTRAN* sequence found in the subroutine `spnprn.f`, with meaning of printed quantities as follows :

LET(I),IEX(I)	:	tagging character and flag (see *OBJET*)
SI(1-4,I)	:	spin components SX, SY, SZ and modulus, at the origin
SF(1-4,I)	:	*idem* at the current position
GAMMA	:	Lorentz relativistic factor
I	:	particle number
IMAX	:	total number of particles ray-traced (see *OBJET*)
IPASS	:	turn number (see *REBELOTE*)

• *SPNSTORE* has an effect similar to *SPNPRNL*, with two more features.

- On the first data line, *FNAME* may be followed by a series of up to 10 *LABEL*s proper to the elements of the zgoubi.dat data file at the exit of which the print should occur ; if no label is given, the print occurs by default at the very location of *SPNSTORE* ; if labels are given, then print occurs right downstream of all optical elements wearing those labels (and no longer at the *SPNSTORE* location).

- The next data line gives a parameter, IP : printing will occur every IP other pass, when using *REBELOTE* with *NPASS* $\geq IP-1$.

For instance the following data input in zgoubi.dat :

```
SPNSTORE
zgoubi.spn      HPCKUP    VPCKUP
12
```

will result in output prints into zgoubi.spn, every 12 other pass, each time elements of the zgoubi.dat data list labeled either *HPCKUP* or *VPCKUP* are encountered.

Note

Binary storage can be obtained from *SPNPRNL* and *SPNSTORE*. This is for the sake of compactness and I/O access speed by zgoubi or zpop, for instance in case voluminous amounts of data should be manipulated. This is achieved by giving the storage file a name of the form *b_FNAME* or *B_FNAME* (*e.g.*, 'b_zgoubi.spn'). The *FORTRAN WRITE* output list is the same as in the *FORMATTED* case above.

5.5.10 SPNPRT : Print spin coordinates

SPNPRT can be introduced anywhere in a structure. It produces a print out (to zgoubi.res) of various informations such as the initial and actual coordinates and modulus of the spin of the *IMAX* particles, their Lorentz factor γ, the mean values of the spin components, etc., at the location where it is placed in the zgoubi.dat data list.

PRINT : If "*PRINT*" appears after *SPNPRT*, then spin data will be stored in zgoubi.SPNPRT.Out. The latter is opened at the first occurence of "*PRINT*". In particular, if *REBELOTE* is used then data will be stacked in zgoubi.SPNPRT.Out.

MATRIX : If "*MATRIX*" appears after *SPNPRT*, then spin rotation matrix (or matrices) will be computed. This is done assuming appropriate initial particle and spin sampling has been defined (by respectively *OBJET* and *SPNTRK*), as follows :

- three particles are needed for *SPNPRT* to compute a spin matrix. All three have identical initial coordinates (for instance, the local stable precession axis coordinates) and rigidity, and their spins are on a direct triedra, for instance, respectively,

$$(S_X, S_Y, S_Z) = (1,0,0),\ (0,1,0),\ (0,0,1),$$

- there may be several groups of three particles, each group with a particular rigidity (*i.e.*, the three particles in a group have the same momentum). OBJET with *KOBJ=1* or *KOBJ=2* can be used for instance to generate these particles by groups of like rigidity. *SPNPRT* will thus compute as many matrices.

5.5.11 SRPRNT : Print SR loss statistics

SRPRNT may be introduced anywhere in a structure. It allows switching on synchrotron radiation energy loss computation. It produces in addition a print out (to zgoubi.res) of current state of statistics on several parameters related to SR loss presumably activated beforehand with keyword *SRLOSS*.

5.5.12 TWISS : Calculation of periodic optical parameters. Log to zgoubi.TWISS.out

TWISS causes the calculation of transport coefficients and various other global parameters, in particular periodical quantities as tunes and optical functions, in the coupled hypothesis. *TWISS* is normally placed at the end of the structure ; it causes a series of up to 5 successive passes in the structure (at the manner of *REBELOTE*).

The object necessary for these calculations will be generated automatically if one uses *OBJET* with option *KOBJ*= 5.

TWISS works in a way similar to *MATRIX*, iterating the *MATRIX* process wherever necessary, changing for instance the reference trajectory in *OBJET* for dp/p related computations. In particular :

- It assumes that the reference particle (particle #1 of 11, when using *OBJET[KOBJ= 5]*) is located on the closed orbit. *This condition has to be satisfied for TWISS to work consistently*, because *TWISS* does not look for the closed orbit by itself (this is under installation).
- A first pass (the only one if *KTW=1*) through the structure allows computing the periodic optical functions at the end of the structure, from the rays.
- The periodic dispersions are used to define chromatic closed orbits at $\pm\delta p/p$. A second and a third pass (which terminate the process if *KTW=2*) with chromatic objects centered respectively on $\pm\delta p/p$ chromatic orbits will then compute the chromatic first order transport matrices. From these the chromaticities are deduced.
- Anharmonicities need two additional passes (which terminate the process if *KTW=3*). They are deduced from the difference in tunes for particles tracked on different transverse invariants, horizontal or vertical.

The execution of *TWISS* will cause printout to the file zgoubi.TWISS.out, a MADX type “twiss” file, with similar formatting, including periodic optical data as a header (tunes, chromaticities, etc.) and, in sequence element by element : optical functions, element strength, first order transport coefficients, spin $\vec{n}_0$ vector (requires *’SPNTRK’* request).

About the source code :

The program `beamat` ensures the transport of the beam matrix along the zgoubi.dat sequence. `beamat` is called by `opticc`, itself called after each optical element, in the program `zgoubi`.

The starting beam matrix value is in the array `FI`, initialized by the program `twiss` when met at the end of the zgoubi.dat sequence. The array `F0` contains the running beam matrix.

The program `optimp`, called from `opticc`, prints to zgoubi.TWISS.out after each optical element of the zgoubi.dat sequence.

5.6 Complements Regarding Various Functionalities

5.6.1 Units in zgoubi

Units to be employed in **zgoubi** input data list (in zgoubi.dat) are stated in Part B of this guide. Part B should be carefully referred to about units to be used, when setting up the data list, as some insight in the code indicates :

Working units in **zgoubi** numerical integrator, and as to the normalized fields delivered to the integrator, satisfy to the following (in particular, whatever the units in the input data file, as listed in Part B, quantities of concern end up manipulated as follows) :

- position coordinates (X, Y, Z, S) in centimeter,
- angle coordinates (T, P) in radian,
- time in μsecond,
- energy in MeV,
- magnetic field in kG
- electrostatic field in MV/cm
- rigidity $B\rho$ in kG.cm,
- fields delivered to the integration procedure after convertion to strengths, units of $m^{-\text{integer}}$, for instance,
  - magnetic dipole strength ($B/B\rho = 1/\rho$) is in m^{-1} (with B the field),
  - 2(n+1) order magnetic multipol strengths ($(B/R^n)/B\rho$, n=1 (quadrupole), n=2 (sextupole), etc.) are in $m^{-(n+1)}$ (with B the field at pole tip radius R),
  - etc.,
  - 2(n+1) order magnetic multipol strengths ($(\Phi/R^{(n+1)})/W$, n=1 (quadrupole), n=2 (sextupole), etc) are in $m^{-(n+1)}$ (with Φ the potential (in Volt) at pole tip radius R and W the kinetic energy of the particle (in electron-Volt)).

For historical reasons, since **zgoubi** was first developed in 1972, implementation of optical elements (as listed in Part B) has resulted in parameter definition resorting to a wide variety of units. For instance :

- element lengths in meter or centimeter,
- positioning angles in rad, mrad, degree,
- magnetic fields in Tesla or kGauss,
- electrostatic potentials in Volt, MVolt,
- electrostatic fields in Volt/m, V/cm, MVolt/m,
- etc.

5.6.2 Reference rigidity

zgoubi computes the strengths of optical elements (they are usually defined by their field) from the reference rigidity *BORO* as defined in *[MC]OBJET*. However using *CAVITE*, and indirectly *SCALING*, may affect the reference rigidity, following

$$B\rho_{ref} = BORO \longrightarrow B\rho_{ref} = BORO + \delta B\rho_s$$

with $\delta B\rho_s$ the synchronous rigidity increase (or decrease). A typical configuration where this would occur is that of multi-turn tracking in a pulsed synchrotron, where in general strengths have to follow the acceleration (see section 5.6.7).

5.6.3 Time Varying Fields

Fields can be varied as a function of time (in some cases this may mean as a function of turn number, see section 5.6.7), by means of the *SCALING* keyword.

Eventually some families of magnets may be given a different timing law for the simulation of special processes (*e.g.*, time varying orbit bump, fast crossing of spin resonances with families of jump quadrupoles).

5.6.4 Backward Ray-Tracing

For the purpose of parameterization for instance, it may be interesting to ray-trace backward from the image toward the object. This can be performed by first reversing the position of optical elements in the structure, and then reversing the integration step sign in all the optical elements.

An illustration of this feature is given in the following Figure 43.

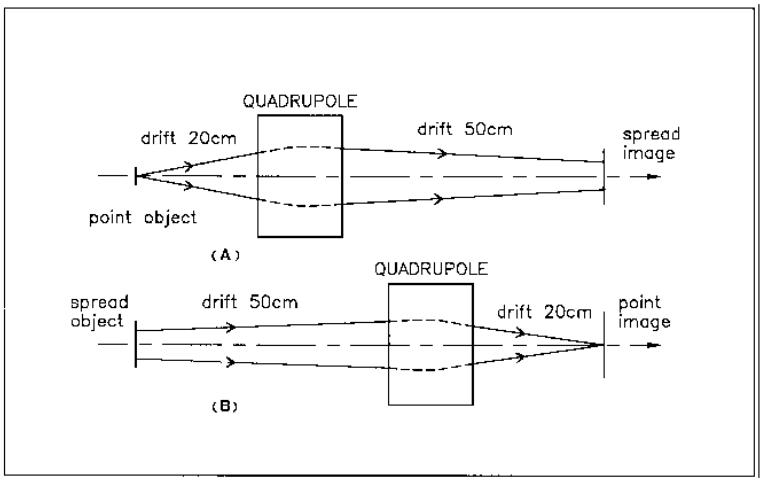


Figure 43: A. Regular forward ray-tracing, from object to image.
B. Same structure, with backward ray-tracing from image to object : negative integration step XPAS is used in the quadrupole.

5.6.5 Checking Fields and Trajectories Inside Optical Elements

• In all optical elements, an option IL is available. It is normally set to $IL = 0$ and in this case has no effect. Other possibilities are as follows :

- $IL = 1$ causes a print in zgoubi.res of particle coordinates, fields and low-order derivatives, spin vector and other data, at each integration step along trajectories across the optical element. In the meantime, a calculation and summation of the values of $\vec{\nabla} \cdot \vec{B}$, $\vec{\nabla} \times \vec{B}$ and $\nabla^2 \vec{B}$ (same for $\vec{E}$) at all integration steps is performed, which allows a check of the behavior of $\vec{B}$ (and/or $\vec{E}$) in field maps (all these derivatives should normally be zero).

- $IL = 2$ causes a print of particle data (coordinates, $\vec{B}$ and/or $\vec{E}$ field(s) experienced, RF cavity related data, spin, etc.), particle status (*e.g.*, decay status, whether it is still tracked, etc.), into the dedicated file zgoubi.plt, one line of data per article, at each integration step. This information can further be processed (using **zpop**[9] or gnuplot, for instance). In order to minimize the volume of that storage file (when dealing with small step size, long optical elements, large number of particles, etc.) it is possible to print out every other 10^n integration step by taking $IL = 2 \times 10^n$ instead (for instance, $IL = 200$ would cause output into zgoubi.plt every 100 other step).

- $IL = 7$ causes similar outputs to $IL = 1$, and more, in particular derivatives to higher order, yet into the dedicated file zgoubi.impdev.out, one line of data per particle, at each integration step. This information can be further plotted (*e.g.*, using **zpop**, or gnuplot). An example is given in fig. 44, a plot of the quadrupole and sextupole field indices along the reference orbit in a combined function main dipole pertaining to the AGS lattice [61].

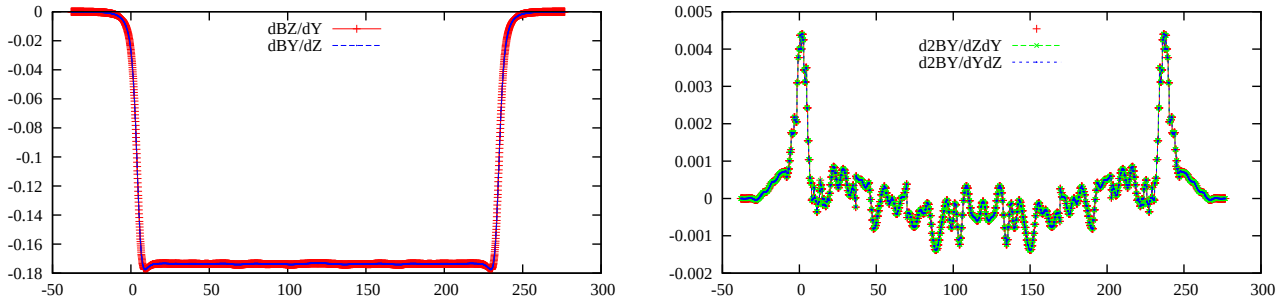


Figure 44: Typical field and derivatives - here, along the 10 GeV proton orbit across an AGS A-type main magnet. Derivatives are obtained from second degree polynomial interpolation from the magnet measured field map. Left : $dB_Z(s)/dY$ and $dB_Y(s)/dZ$, right : $d^2B_Y(s)/dZdY$, $d^2B_Y(s)/dYdZ$. These data were stored in zgoubi.impdev.out during execution, by stating $IL = 7$ under *TOSCA* keyword.

• When dealing with field maps (*e.g.*, *CARTEMES*, *ELREVOL*, *TOSCA*), an option index IC is available. It is normally set to $IC = 0$ and in this case has no effect. Other possibilities are as follows :

- $IC = 1$ causes a print of the field map in zgoubi.res.

- $IC = 2$ will cause a print of the field map into the file zgoubi.map which can further be processed with **zpop** for plotting, superimposing trajectories, and other data treatment purposes.

5.6.6 Labeling Keywords

Keywords in **zgoubi** input data file zgoubi.dat can be *LABEL*'ed, for the purpose of the execution of such procedures as *REBELOTE*, *PICKUPS*, *FAISCNL*, *FAISTORE*, *SCALING*, and also for the purpose of particle coordinate storage into zgoubi.plt (see section 5.6.5, $IL = 2$ option).

[9] See Part D of the Guide.

A keyword in zgoubi.dat accepts two *LABEL*s. The first one is used for the above mentioned purposes. An instance of the use of the second one is with *MARKER* : it can be set to ".plt" for storage of current particle data into zgoubi.plt. The keyword and its *LABEL*[s] should fit within a 110-character long string on a single line (a quantity set in the *FORTRAN* file `prdata.f`).

5.6.7 Multi-turn Tracking in Circular Machines

Multi-turn tracking in circular machines can be performed by means of the keyword *REBELOTE*. *REBELOTE* is introduced in the zgoubi.dat data list with its argument *NPASS*$+1$ being the number of turns to be performed. It will cause a jump of the "multi-turn pointer" back to (details on page 77), either the beginning of the data list (default case), or to a particular *LABEL* in that list. From then on, tracking resumes down to *REBELOTE* again, and so forth until the requested number of passes has been reached.

Possible magnet timing laws $B(T)$ during the multi-turn process (with $T = 1$ to *NPASS*$+1$ counted in number of turns) can be introduced by means of *SCALING*.

In order that the *IMAX* particles of the beam start a new pass with the coordinates they had reached at the end of the previous one, the option $K = 99$ has to be specified in *REBELOTE*.

Synchrotron acceleration can be simulated, using following the procedure :

- *CAVITE* appears in the zgoubi.dat data list (normally before *REBELOTE*), with option *IOPT*$\neq 0$,
- the RF frequency of the cavity may be given a timing law $f_{RF}(T)$ by means of *SCALING*, family *CAVITE*,
- the magnets are given a field timing law $B(T)$, (with $T = 1$ to *NPASS*$+1$ counted in number of turns) by means of *SCALING*.

Eventually some families of magnets may be given a different timing law for the simulation of special processes (*e.g.*, time varying orbit bump, fast crossing of spin resonances with families of jump quadrupoles).

5.6.8 Positioning, (Mis-)Alignment, of Optical Elements and Field Maps

The last record in most optical elements and field maps is the positioning option *KPOS*. *KPOS* is followed by the positioning parameters, *e.g.*, *XCE, YCE* for translation and *ALE* for rotation. The positioning works in two different ways, depending whether the element is defined in Cartesian (X, Y, Z) coordinates (*e.g.*, *QUADRUPO, TOSCA*), or polar (R, θ, Z) coordinates (*DIPOLE*).

Cartesian Coordinates :

If *KPOS* $= 1$, the optical element is moved (shifted by *XCE, YCE* and Z-rotated by *ALE*) with respect to the incoming reference frame. Trajectory coordinates after traversal of the element refer the element frame.

If *KPOS*$= 2$, the shifts *XCE* and *YCE*, and the tilt angle *ALE* are taken into account, for mis-aligning the element with respect to the incoming reference, as shown in Fig. 45. The effect is equivalent to a *CHANGREF(XCE, YCE, ALE)* upstream of the optical element, followed by *CHANGREF(XCS,YCS,ALS* $= -$*ALE)* downstream of it, with computed *XCS*, *YCS* values as schemed in Fig. 45.

KPOS $= 3$ option is available for some optical elements (*e.g.*, *BEND, MULTIPOL, AGSMM*) ; it is effective only if a non zero dipole component $B1$ is present, or if *ALE* is non-zero. It positions automatically the magnet in a symmetric manner with respect to the incoming and outgoing reference axis, convenient for periodic structures, as follows (Fig 46).

Both incoming and outgoing reference frames are tilted w.r.t. the magnet,

- either, by an angle *ALE* if ALE$\neq$0,
- or, if ALE=0 by half the *Z*-rotation $\alpha_Z/2$ such that $L = 2\dfrac{BORO}{B1}\sin(\alpha_Z/2)$, wherein L = geometrical length, *BORO* = reference rigidity as defined in *OBJET*.

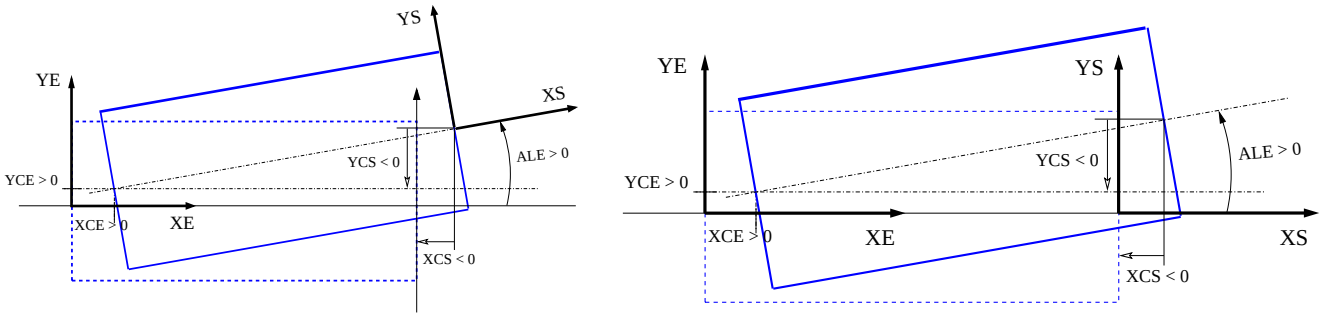


Figure 45: Case of Cartesian frame optical element. Left : positioning an optical element using *KPOS*$= 1$. Right : Mis-aligning an optical element using *KPOS*$= 2$. (X_E, Y_E) and (X_S, Y_S) are respectively the incoming and outgoing reference frames.

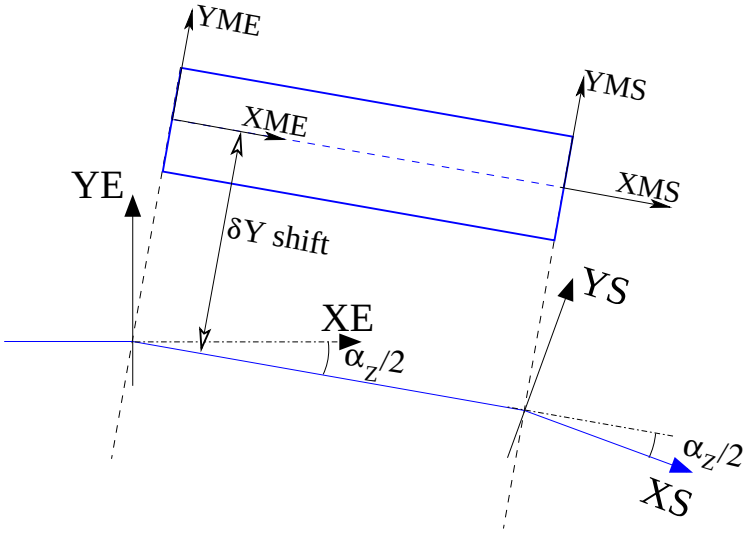


Figure 46: Alignment in *BEND* by half the deviation, $\alpha_Z/2$, using *KPOS*$= 3$. (X_E, Y_E) and (X_S, Y_S) are respectively the incoming and outgoing reference frames. Trajectory coordinates inside the magnet are defined in the magnet frame, (X_{ME}, Y_{ME}).

Next, the optical element is Y-shifted by $\delta Y = YCE$ (*XCE* is not used) in a direction orthogonal to the new magnet axis (*i.e.*, at an angle $ALE + \pi/2$ *wrt.* the X axis of the incoming reference frame).

KPOS $= 4$ applies to *AGSMM* (AGS main magnet) and *MULTIPOL*. It orients the magnet in a way similar to *KPOS* $= 3$, with reference frame Z-rotated by $\alpha_Z/2$ as drawn from $L = 2\dfrac{BORO}{B1}\sin(\alpha_Z/2)$.
However additional magnet mis-alignments (alignment errors) are available (Fig. 47) : Z-rotation θ_Z, Z-shift δZ, Y-rotation θ_Y. Rotations apply at the center of the magnet.

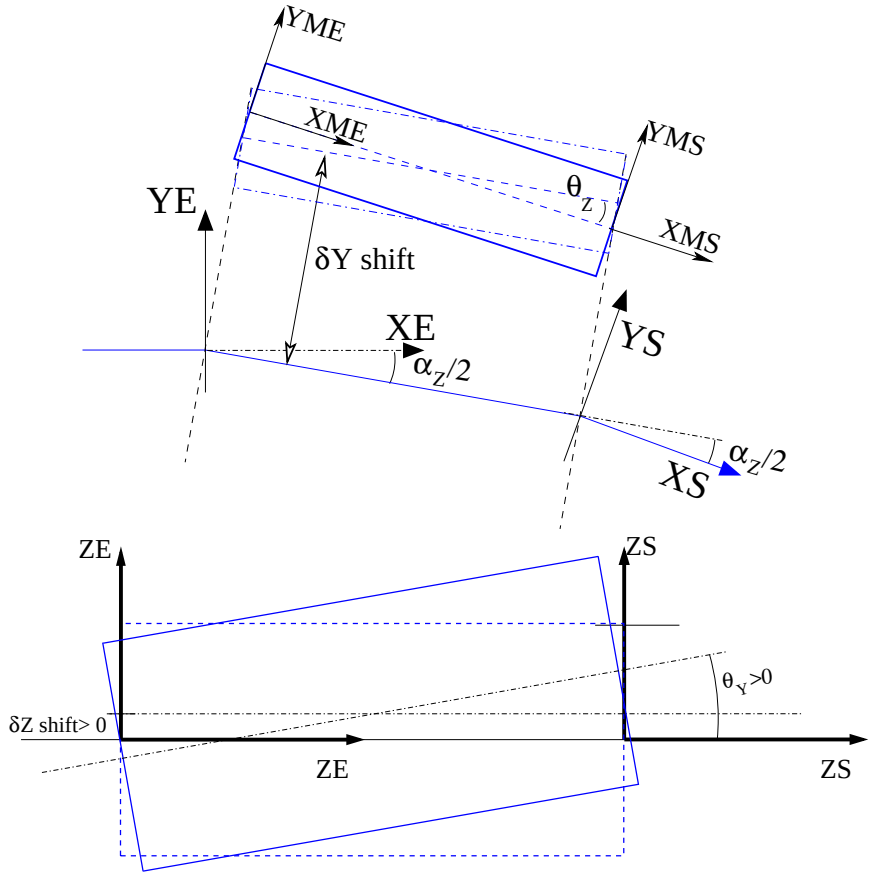


Figure 47: Additional magnet mis-alignement with *KPOS=4.*

Polar Coordinates

If *KPOS* $= 1$, the element is positioned automatically in such a way that a particle entering with zero initial coordinates and $1 + DP = B\rho/$*BORO* relative rigidity will reach position (RM, $\frac{AT}{2}$) in the element with $T = 0$ angle with respect to the moving frame in the polar coordinates system of the element (Fig. **??** ; see *DIPOLE-M* and *POLARMES*).
If *KPOS* $= 2$, the map is positioned in such a way that the incoming reference frame is presented at radius RE with angle TE. The exit reference frame is positioned in a similar way with respect to the map, by means of the two parameters RS (radius) and TS (angle) (see Fig. 11A page 92A).

5.6.9 Coded Integration Step

In several optical elements (*e.g.*, all multipoles, *BEND*) the integration step (in general noted *XPAS*) can be coded under the form *XPAS* = #E|C|S, with E, C, S integers, E is the number of steps in the entrance fringe field region, C is the number of steps in the magnet body, and S is the number of steps in the exit fringe field region.

5.6.10 Ray-tracing of an Arbitrarily Large Number of Particles

Monte Carlo multi-particle simulations involving an arbitrarily large number of particles can be performed using *MCOBJET* together with *REBELOTE*, put at the end of the optical structure, with its argument *NPASS* being the number of passes through *REBELOTE*, and (*NPASS*$+1$) * *IMAX* the number of particles to be ray-traced. In order that new initial conditions (D, Y, T, Z, P, X) be generated at each pass, $K = 0$ has to be specified in *REBELOTE*.
Statistics on coordinates, spins, and other histograms can be performed by means of such procedures as *HISTO, SPNTRK*, etc. that stack the information from pass to pass.

5.6.11 Stopped Particles : The *IEX* Flag

As described in *OBJET*, each particle $I = 1$, *IMAX* is attached a value *IEX*(I) of the *IEX* flag. Normally, *IEX*$(I) = 1$. Under certain circumstances, *IEX* may be changed to a negative value by **zgoubi**, as follows.

-1 : the trajectory happened to wander outside the limits of a field map

-2 : too many integration steps in an optical element (a quantity controlled in MXSTEP.H include file)

-3 : deviation happened to exceed $\pi/2$ in an optical element not designed to allow that

-4 : stopped by walls (procedures *CHAMBR, COLLIMA*)

-5 : too many iterations in subroutine depla.f

-6 : energy loss exceeds particle energy

-7 : field discontinuities larger than 50% within a field map

-8 : reached field limit in an optical element

Only in the case *IEX* $= -1$ will the integration not be stopped since in this case the field outside the map is extrapolated from the map data, and the particle may possibly get back into the map (see section 1.4.2 on page 29). In all other cases the particle will be stopped.

5.6.12 Negative Rigidity

zgoubi can handle negative rigidities $B\rho = p/q$. This is equivalent to considering either particles with negative charge ($q < 0$) or momentum ($p < 0$), or reversed fields (*wrt.* the field sign that shows in the zgoubi.dat optical element data list).
Negative rigidities may be specified in terms of *BORO* < 0 or $D = B\rho/$*BORO* < 0 when defining the initial coordinates with *OBJET* or *MCOBJET*.

PART B

Keywords and input data formatting

Glossary of Keywords, Part B

Available keywords : how to format their input data list.

Optical elements versus keywords

What can be simulated
What keyword(s) can be used for that

This glossary gives a list of keywords suitable for the simulation of common optical elements. These are classified in three categories: magnetic, electric and combined electro-magnetic elements.

Field map procedures are also listed; they provide a means for ray-tracing through measured or simulated electric and/or magnetic fields.

MAGNETIC ELEMENTS

AGS main magnet	AGSMM
Cyclotron magnet or sector	DIPOLE[S], DIPOLE-M, FFAG, FFAG-SPI
Decapole	DECAPOLE, MULTIPOL
Dipole[s], spectrometer dipole	AIMANT, BEND, DIPOLE[S][-M], MULTIPOL, QUADISEX
Dodecapole	DODECAPO, MULTIPOL
FFAG magnets	DIPOLES, FFAG, FFAG-SPI, MULTIPOL
Helical dipole	HELIX
Multipole	MULTIPOL, QUADISEX, SEXQUAD
Octupole	OCTUPOLE, MULTIPOL, QUADISEX, SEXQUAD
Quadrupole	QUADRUPO, MULTIPOL, SEXQUAD, AGSQUAD
Sextupole	SEXTUPOL, MULTIPOL, QUADISEX, SEXQUAD
Skew multipoles	MULTIPOL
Solenoid	SOLENOID
Undulator	UNDULATOR

Using field maps

1-D, cylindrical symmetry	BREVOL
2-D, mid-plane symmetry	CARTEMES, POISSON, TOSCA
2-D, no symmetry	MAP2D
2-D, polar mesh, mid-plane symmetry	POLARMES
3-D, no symmetry	TOSCA
EMMA FFAG quadrupole doublet	EMMA
linear composition of field maps	TOSCA

ELECTRIC ELEMENTS

2-tube (bipotential) lens	EL2TUB
3-tube (unipotential) lens	UNIPOT
Decapole	ELMULT
Dipole	ELMULT
Dodecapole	ELMULT
Multipole	ELMULT
N-electrode mirror/lens, straight slits	ELMIR
N-electrode mirror/lens, circular slits	ELMIRC
Octupole	ELMULT
Quadrupole	ELMULT
R.F. (kick) cavity	CAVITE
Sextupole	ELMULT
Skew multipoles	ELMULT

Using field maps

1D, cylindrical symmetry	ELREVOL

2-D, no symmetry	MAP2D-E

ELECTRO-MAGNETIC ELEMENTS

Decapole	EBMULT
Dipole	EBMULT
Dodecapole	EBMULT
Multipole	EBMULT
Octupole	EBMULT
Quadrupole	EBMULT
Sextupole	EBMULT
Skew multipoles	EBMULT
Wien filter	SEPARA, WIENFILT

INTRODUCTION

Here after is given a detailed description of input data formatting and units. All available keywords appear in alphabetical order.

Keywords are read from the input data file by an unformatted *FORTRAN READ* statement. They be enclosed between quotes (*e.g.*, '*DIPOLE'*).

Text string data such as comments or file names, are read by formatted READ statements, no quotes should be used in that case.

Numerical variables and indices are read by unformatted READ. It may therefore be necessary that integer variables be assigned an integer value.

In the following tables

- the first column shows the expected input parameters (actually, their *values* are expected), indices and text strings,
- the second column gives brief comments regarding their meaning and use,
- the third column gives the units or ranges,
- the fourth column indicates whether the expected parameter types are integer (I), real (E) or text string (A). For example, "I, 3*E" means that one integer followed by 3 reals is expected. "A80" means that a text string of maximum 80 characters is expected.

AGSMM	**AGS main magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
MOD, dL, R_0, $dB1$, $dB2$, $dB3$	Type of magnet model [1] ; unused ; pole tip radius, 10 cm if set to zero ; relative error on dipole, quadrupole, sextupole component.	2*no dim., cm, 3*no dim.	I[.I], 5*E
NBLW[.MOD], *NBLW times :* NW, I	Number of back-leg windings [back-leg winding model [2]] ; for each back-leg winding : number of windings, current.	$\leq$2, *NBLW*$\times$ (any, Amp.)	I, *NBLW*$\times$ (I, E)
	Entrance face		
X_E, λ_E, E_2, E_3	Integration zone ; fringe field extent : dipole fringe field extent = λ_E ; quadrupole fringe field extent = $\lambda_E * E_2$; sextuppole fringe field extent = $\lambda_E * E_3$ (sharp edge if field extent is zero)	2*cm, 2*no dim.	4*E
NCE, $C_0 - C_5$	same as *QUADRUPO*	0-6, 6*no dim.	I, 6*E
	Exit face		
X_S, λ_S, S_2, S_3	Integration zone ; as for entrance	2*cm, 2*no dim.	4*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
$R1$, $R2$, $R3$	Skew angles of field components	3*rad	10*E
XPAS	Integration step	cm	E
If KPOS = 1-3 *KPOS, XCE, YCE, ALE*	Positioning as follows : *KPOS*=1 : element aligned ; *KPOS*=2 : misaligned ; shifts, tilt ; *KPOS*=3 : effective only if $B1 \neq 0$: entrance and exit frames are tilted *wrt.* the magnet by an angle of • either ALE if ALE$\neq$0 • or 2 Arcsin$(B1\ XL\ /\ 2BORO)$ if ALE=0	1-3, 2*cm, rad	I, 3*E
If KPOS = 4			
KPOS, XS, YS, ZR, ZS, YR	X-, Y-shift, Z-rotation, Z-shift, Y-rotation Automatic ALE positioning as *KPOS* = 3, and in addition possible X- or Y- or Z-misalignment or Z-, Y-rotation. (under development).	4, cm, 2*(cm, rad)	I, 5*E

[1] *MOD=1* : centered multipole model ; *MOD=2* : long-shifted dipole model ; *MOD=3* : short-shifted dipole model.

[2] *MOD = 0* (default) : user defined back-leg windings (defined in routine `agsblw.f`) ; *MOD = 1* : actual AGS data are taken, namely : MM_A16AD : NBLW = 1, SIGN = 1.D0, NW = 10 ; MM_A17CF : NBLW = 1, SIGN = 1.D0, NW = 10 ; MM_A18CF : NBLW = 1, SIGN = -1.D0, NW = 10 ; MM_A19BD : NBLW = 1, SIGN = -1.D0, NW = 12 ; MM_A20BD : NBLW = 1, SIGN = 1.D0, NW = 12 ; MM_B02BF : NBLW = 2, SIGN = 1.D0, NW1 = 12, SIGN = 1.D0, NW2 = 6 ; MM_B03CD : NBLW = 1, SIGN = 1.D0, NW = 10 ; MM_B04CD : NBLW = 1, SIGN = -1.D0, NW = 10 ; MM_B05A : NBLW = 1, SIGN = -1.D0, NW = 10 ; MM_K19BD : NBLW = 1, SIGN = 1.D0, NW = 6 ; MM_K20B : NBLW = 1, SIGN = 1.D0, NW = 6 ; MM_L13CF : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_L14C : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_A07CD : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_A08C : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_B01B : NBLW = 1, SIGN = 1.D0, NW = 6 ; MM_L06A : NBLW = 1, SIGN = 1.D0, NW = 5 ; MM_L07C : NBLW = 1, SIGN = 1.D0, NW = 5 ; MM_A14C : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_A15A : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_E06A : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_E07CD : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_E20BD : NBLW = 1, SIGN = 1.D0, NW = 6 ; MM_F01BF : NBLW = 1, SIGN = 1.D0, NW = 6 ; MM_F14CF : NBLW = 1, SIGN = 1.D0, NW = 5 ; MM_F15AD : NBLW = 1, SIGN = 1.D0, NW = 5 ; MM_G08CD : NBLW = 1, SIGN = -1.D0, NW = 5 ; MM_G09BF : NBLW = 1, SIGN = -1.D0, NW = 6.

AGSQUAD **AGS quadrupole**

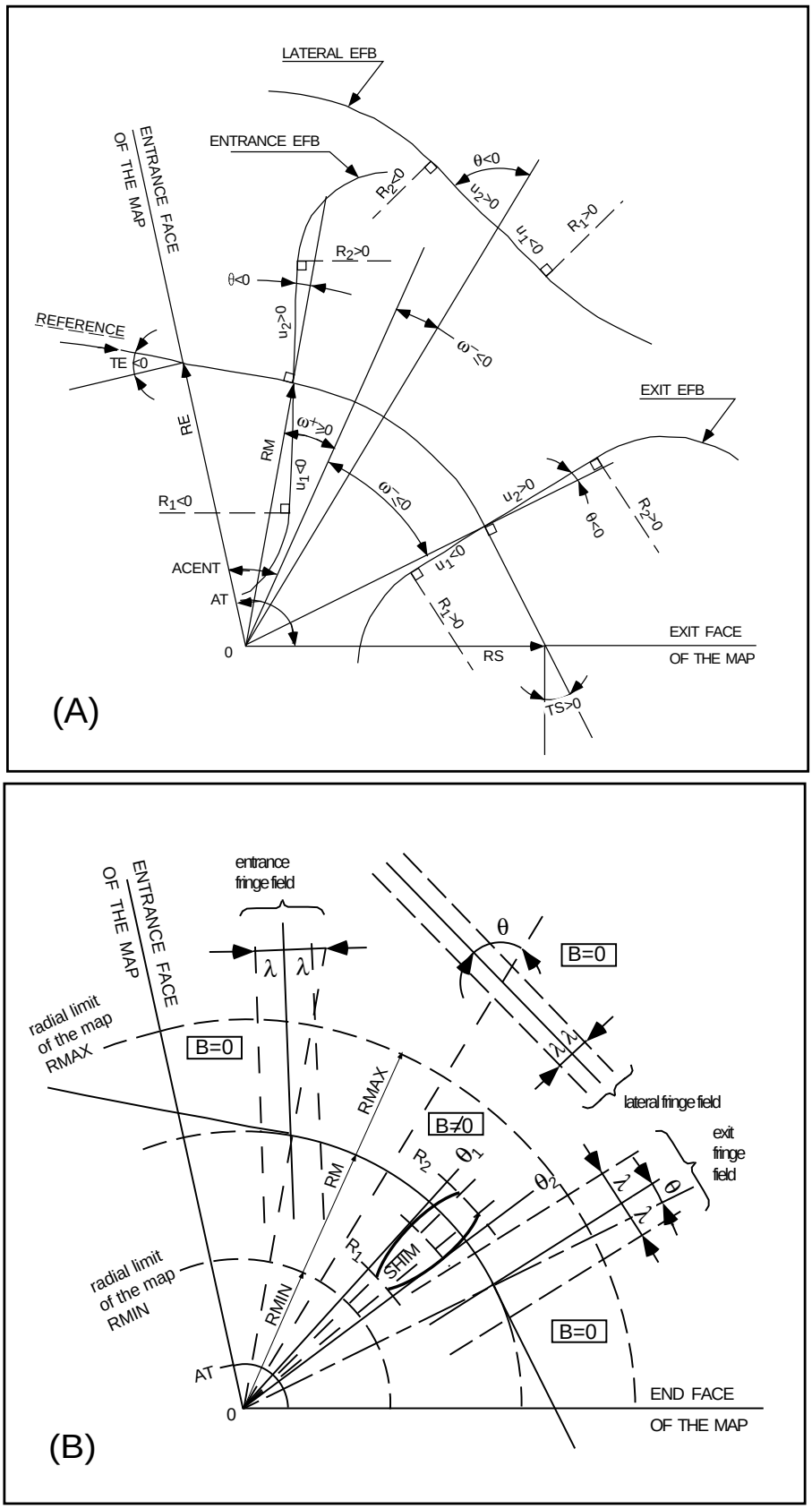
		dim.	data type	FIT numb.
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I	
XL, R_0, I_{W1}, I_{W2}, I_{W3}, dI_{W1}, dI_{W2}, dI_{W3}	Length of element ; radius at pole tip ; current in windings ; relative error on currents.	2*cm, 3*A 3*no dim	5*E 3*E	 10 - 17
	Entrance face			
X_E, λ_E	Integration zone ; fringe field extent. (sharp edge if field extent is zero)	2*cm	11*E	20, 21
NCE, $C_0 - C_5$	Same as *QUADRUPO*	0-6, 6*no dim.	I, 6*E	30 - 36
	Exit face			
X_S, λ_S	Integration zone ; as for entrance	2*cm	11*E	40, 41
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E	50 - 56
$R1$	Roll angle	rad	E	60
XPAS	Integration step	cm	E	70
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E	80 - 83

AIMANT **Generation of dipole mid-plane 2-D map, polar frame**

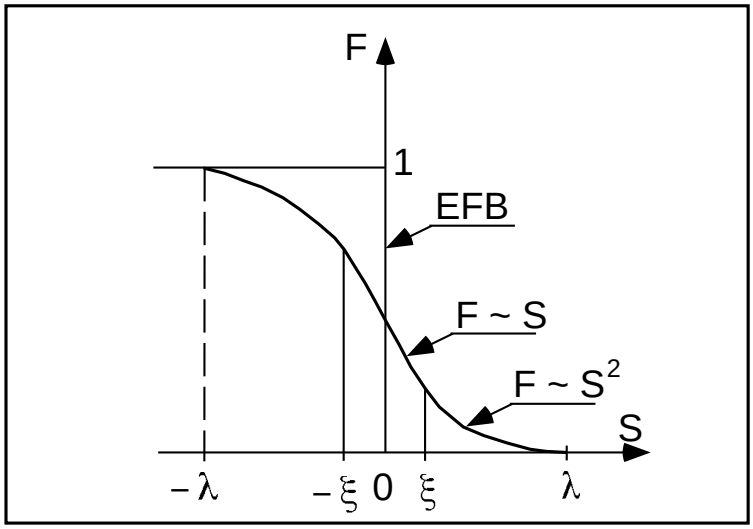
$$B_Z = \mathcal{F}B_0\left(1 - N\left(\frac{R-RM}{RM}\right) + B\left(\frac{R-RM}{RM}\right)^2 + G\left(\frac{R-RM}{RM}\right)^3\right)$$

NFACE, IC, IL	Number of field boundaries $IC = 1, 2$: print field map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	2-3; 0-2; 0-2$[\times 10^n]$, 7	3*I
IAMAX, IRMAX	Azimuthal and radial number of nodes of the mesh	≤ 400, $\leq 10^4$	2*I
B_0*, N, B, G*	Field and field indices	kG, 3*no dim.	4*E
AT, ACENT, RM, RMIN, RMAX	Mesh parameters : total angle of the map ; azimuth for EFBs positioning ; reference radius ; minimum and maximum radii	2*deg, 3*cm	5*E
	ENTRANCE FIELD BOUNDARY		
λ, ξ	Fringe field extent (normally $\simeq$ gap size) ; flag : - if $\xi \geq 0$: second order type fringe field with linear variation over distance ξ - if $\xi = -1$: exponential type fringe field : $F = (1 + \exp(P(s)))^{-1}$ $P(s) = C_0 + C_1(\frac{s}{\lambda}) + C_2(\frac{s}{\lambda})^2 + ... + C_5(\frac{s}{\lambda})^5$	cm, (cm)	2*E
NC, $C_0 - C_5$*, shift*	NC = 1 + degree of $P(s)$; C_0 to C_5 : see above ; EFB shift (ineffective if $\xi \geq 0$)	0-6, 6*no dim., cm	I, 7*E
ω^+, θ, R_1, U_1, U_2, R_2	Azimuth of entrance EFB with respect to *ACENT* ; wedge angle of EFB ; radii and linear extents of EFB (use $U_{1\atop 2} = \mp\infty$ when $R_{1,2} = \infty$)	2*deg, 4*cm	6*E
	(Note : $\lambda = 0$, ω^+ = ACENT and $\theta = 0$ for sharp edge)		
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
λ, ξ	Fringe field parameters	cm, (cm)	2*E
NC, $C_0 - C_5$*, shift*	See above	0-6, 6*no dim., cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2	Positioning and shape of the exit EFB	2*deg, 4*cm	6*E
	(Note : $\lambda = 0$, ω^- =-AT+ACENT and $\theta = 0$ for sharp edge)		

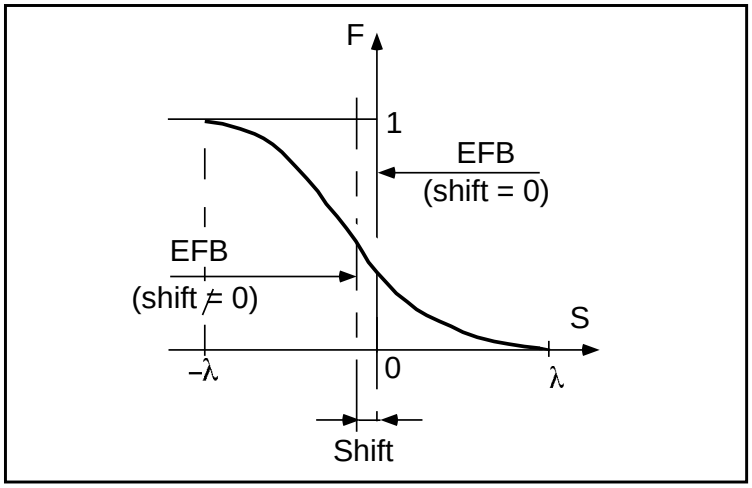
If NFACE = 3	LATERAL FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY) Next 3 records *only* if *NFACE* = 3		
λ, ξ	Fringe field parameters	cm, (cm)	2*E
NC, $C_0 - C_5$, *shift*	See above	0-6, 6*no dim., cm	I, 7*E
ω^-, θ, R_1, U_1, U_2, R_2, $RM3$	Positioning and shape of the lateral EFB ; RM3 is the radial position on azimuth *ACENT*	2*deg, 5*cm	7*E
NBS	Option index for perturbations to the field map	-2 – 0 or ≥ 1	I
If NBS = 0	Normal value. No other record required		
If NBS = -2	The map is modified as follows :		
R_0, $\Delta B/B_0$	B transforms to $B * \left(1 + \frac{\Delta B}{B_0} \frac{R - R_0}{RMAX - RMIN}\right)$	cm, no dim.	2*E
If NBS = -1	the map is modified as follows :		
θ_0, $\Delta B/B_0$	B transforms to $B * \left(1 + \frac{\Delta B}{B_0} \frac{\theta - \theta_0}{AT}\right)$	deg, no dim.	2*E
If NBS ≥ 1	Introduction of NBS shims		
For I = 1, NBS	The following 2 records must be repeated NBS times		
R_1, R_2, θ_1, θ_2, λ	Radial and angular limits of the shim ; λ is unused	2*cm, 2*deg, cm	5*E
γ, α, μ, β	geometrical parameters of the shim	2*deg, 2*no dim.	4*E
IORDRE	Degree of interpolation polynomial : 2 = second degree, 9-point grid 25 = second degree, 25-point grid 4 = fourth degree, 25-point grid	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS	Positioning of the map, normally 2. Two options :	1-2	I
If KPOS = 2	Positioning as follows :		
RE, TE, RS, TS	Radius and angle of reference, respectively, at entrance and exit of the map.	cm, rad, cm, rad	4*E
If KPOS = 1	Automatic positioning of the map, by means of		
DP	reference relative momentum	no dim.	E



A : Parameters used to define the field map and geometrical boundaries.
B : Parameters used to define the field map and fringe fields.



Second order type fringe field.



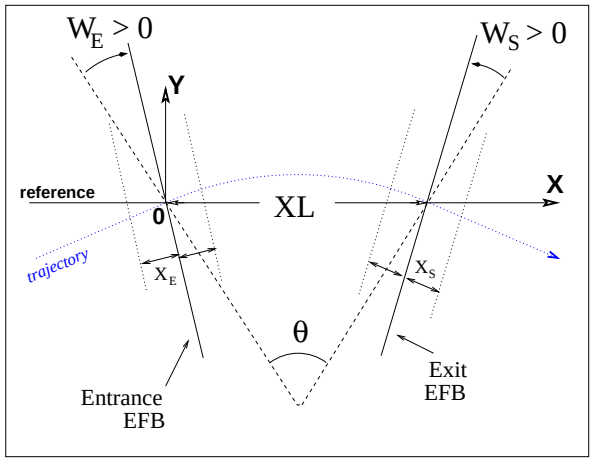
Exponential type fringe field.

AUTOREF	**Transport beam into a new reference frame**		
I	1 : Equivalent to *CHANGREF*($XCE = 0$, $YCE = Y(1)$, $ALE = T(1)$), *i.e.*, recentering of the beam on particle #1.	1-2 1-2	I I
	2 : Equivalent to *CHANGREF*(XW, YW, $T(1)$), with (XW, YW) the location of the intersection (waist) of particles 1, 4 and 5 (useful with *MATRIX*, for automatic positioning of the first order focus).		
	3 : Equivalent to *CHANGREF*(XW, YW, $T(I1)$), with (XW, YW) the location of the intersection (waist) of particles $I1$, $I2$ and $I3$ (for instance : $I1 =$ central trajectory, $I2$ and $I3 =$ paraxial trajectories that intersect at the first order focus).		
	4 : Equivalent to *CHANGREF*(XCE, YCE, ALE). 4.1 : Equivalent to *CHANGREF*(XCE, YCE, ALE) with in addition centering of the beam on a new relative momentum *DCE*.		
	5 : The beam is centered vertically on *ZCE, PLE*.		
If $I = 3$ $I1$, $I2$, $I3$	Provide next record if $I = 3$ Three particle numbers	 3*(1-*IMAX*)	 3*I
If $I = 4$ *XCE, YCE, ALE*	Provide next record if $I = 4$ *XCE* and beam centroid new coordinates *YCE, ALE*	 2*cm, mrad	 3*E
If $I = 4.1$ *XCE, YCE, ALE, DCE, TIME*	Provide next record if $I = 4.1$ *XCE* and beam centroid new coordinates *YCE, ALE, DCE, TIME*	 2*cm, mrad, -, μs	 6*E
If $I = 4.2$ *XCE, YCE, ALE, DCE, TIME*	Provide next record if $I = 4.2$ *XCE* and beam centroid new coordinates *YCE, ALE, DCE,* time setting *TIME* (same for all particles).	 2*cm, mrad, -, μs	 6*E
If $I = 5$ *ZCE, PLE*	Provide next record if $I = 5$ New vertical beam centroid coordinates *ZCE, PLE*	 cm, mrad	 2*E

BEAMBEAM **Beam-beam lens**

SW, I	0/1 : off/on ; beam intensity. Use *SPNTRK* to activate spin kicks.	0-2, Amp	I, E
$\alpha_Y,\ \beta_Y,\ \epsilon_{Y,norm}/\pi$	Beam parameters, horizontal.	- , m, m.rad	3*E
$\alpha_Z,\ \beta_Z,\ \epsilon_{Z,norm}/\pi$	Beam parameters, vertical.	- , m, m.rad	3*E
$\sigma_X,\ \sigma_{dp/p}$	*rms* bunch length ; *rms* momentum spread.	m, -	2*E
$\mathcal{C},\ \alpha$	Ring circumference ; momentum compaction.	m, -	2*E
$Q_Y,\ Q_Z,\ Q_s$	Tunes, horizontal, vertical, synchrotron.	-, -, -	3*E
$A_Y,\ A_Z,\ A_X$	Amplitudes, horizontal, vertical, longitudinal.	-, -, -	3*E

BEND	**Bending magnet, Cartesian frame**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, Sk, $B1$	Length ; skew angle ; field (change ALE and wedge signs, if B<0).	cm, rad, kG	3*E
	Entrance face :		
$X_{\rm E}$, $\lambda_{\rm E}$, $W_{\rm E}$	Integration zone extent ; fringe field extent (normally $\simeq$ gap height ; zero for sharp edge) ; wedge angle (>0 for rectangle magnet).	cm, cm, rad	3*E
N, C_0–C_5	Unused ; fringe field coefficients : $B(s) = B1\, F(s)$ with $F(s) = 1/(1+\exp(P(s))$ and $P(s) = \sum_{i=0}^{5} C_i (s/\lambda)^i$	unused, 6*no dim.	I, 6*E
	Exit face :		
X_S, λ_S, W_S	See entrance face	cm, cm, rad	3*E
N, C_0–C_5		unused, 6*no dim.	I, 6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, Z-tilt. *KPOS* = 3 : (*XCE* is unused) entrance and exit frames are shifted by *YCE* and tilted *wrt.* the magnet by an angle of • either ALE, if ALE is non-zero (normally, ALE>0 in that case) • or 2 Arcsin($B1 \times XL\ /\ 2 \times BORO$) if ALE=0	1-2, 2*cm, rad	I, 3*E



Geometry and parameters of *BEND* : *XL* = length, θ = deviation, W_E, W_S are the entrance and exit wedge angles. The motion is computed in the Cartesian frame (O, X, Y, Z)

• ***Example*** : Positive or negative field *BEND*

The data list at left hand, below, simulates a case of positive field, followed by two different ways to simulate the same magnet with negative field. (This can be copy-pasted to **zgoubi**, as is, for execution.)
The right hand column shows the transport matrix in all three cases.

```
Test signs, with wedges.

! Magnet deviation is
! tta=11.25deg=2asin(L/2rho)=2asin(B.L/2.Brho) with L=247.30039, Brho=1000, B=1.57776

! 1/ Positive B, regular form of input data
 'OBJET'
1000.
5
.01 .1 .01 .1 0. .001
0. 0. 0. 0. 0. 1.

 'BEND'
0
247.30039  0.  1.57776
20. 8. 0.04276056667                          ! wedge is +2.45deg
4 .2401  1.8639  -.5572  .3904 0. 0. 0.
20. 8. 0.04276056667                          ! wedge is +2.45deg
4 .2401  1.8639  -.5572  .3904 0. 0. 0.
#30|10|30
3 0. 0. -.1963495408

 'FAISCEAU'
 'MATRIX'
1 0

! 2/ Negative B. Compared to 1/ : R11-R44 unchaged, R16, R26 change sign.
! 2-a/ By changing sign of B. Note : signs of ALE and wedges have to be reversed.
 'OBJET'
1000.
5
.01 .1 .01 .1 0. .001
0. 0. 0. 0. 0. 1.

 'BEND'
0
247.30039  0.  -1.57776
20. 8. -0.04276056667                          ! wedge is +2.45deg
4 .2401  1.8639  -.5572  .3904 0. 0. 0.
20. 8. -0.04276056667                          ! wedge is +2.45deg
4 .2401  1.8639  -.5572  .3904 0. 0. 0.
#30|10|30
3 0. 0. .1963495408

 'FAISCEAU'
 'MATRIX'
1 0

! 2-a/ Using YMY instead. All data in BEND remain unchanged.
 'OBJET'
1000.
5
.01 .1 .01 .1 0. .001
0. 0. 0. 0. 0. 1.

 'YMY'
 'BEND'
0
247.30039  0.  1.57776
20. 8. 0.04276056667                          ! wedge is +2.45deg
4 .2401  1.8639  -.5572  .3904 0. 0. 0.
20. 8. 0.04276056667                          ! wedge is +2.45deg
4 .2401  1.8639  -.5572  .3904 0. 0. 0.
#30|10|30
3 0. 0. -.1963495408
 'YMY'

 'FAISCEAU'
 'MATRIX'
1 0

 'END'
```

```
Reference particle (#     1), path length :   248.88547

        TRANSFER  MATRIX  ORDRE  1  (MKSA units)
 0.940559         2.42528         0.00000         0.00000  0.0   0.482374
-4.768802E-02     0.940229        0.00000         0.00000  0.0   0.385904
 0.00000          0.00000         0.985447        2.48908  0.0   0.00000
 0.00000          0.00000        -1.163098E-02    0.985390 0.0   0.00000
 0.385970         0.482384        0.00000         0.00000  1.0   6.370064E-02
 0.00000          0.00000         0.00000         0.00000  0.0   1.00000
```

```
Reference particle (#     1), path length :   248.88547

        TRANSFER  MATRIX  ORDRE  1  (MKSA units)
 0.940559         2.42528         0.00000         0.00000  0.0  -0.482374
-4.768802E-02     0.940229        0.00000         0.00000  0.0  -0.385904
 0.00000          0.00000         0.985447        2.48908  0.0   0.00000
 0.00000          0.00000        -1.163098E-02    0.985390 0.0   0.00000
-0.385970        -0.482384        0.00000         0.00000  1.0   6.370064E-02
 0.00000          0.00000         0.00000         0.00000  0.0   1.00000
```

```
Reference particle (#     1), path length :   248.88547

        TRANSFER  MATRIX  ORDRE  1  (MKSA units)
 0.940559         2.42528         0.00000         0.00000  0.0  -0.482374
-4.768802E-02     0.940229        0.00000         0.00000  0.0  -0.385904
 0.00000          0.00000         0.985447        2.48908  0.0   0.00000
 0.00000          0.00000        -1.163098E-02    0.985390 0.0   0.00000
-0.385970        -0.482384        0.00000         0.00000  1.0   6.370064E-02
 0.00000          0.00000         0.00000         0.00000  0.0   1.00000
```

BINARY ***BINARY/FORMATTED* data converter**

$NF[.J],\ NCol,\ NHDR$	Number of files to convert [READ format type, see below], of data columns, of header lines.	$\leq 20, \leq 7\ , 0-9$	3*I1

The next NF lines :

FNAME	Name of the file to be converted. File content is assumed binary *iff* name begins with "B_" or "b_", assumed formatted otherwise.	A80

READ format, case of formatted input file :

If FRM not given	Format is '*'
If FRM=1	Format is '1X, 7E11.*'

READ format, case of binary input file :

Expected format is 7 column rows.

BREVOL	**1-D uniform mesh magnetic field map** X-axis cylindrical symmetry is assumed		
IC, IL	$IC = 1, 2$: print the map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
BNORM, XN	Field and X-coordinate normalization coeff. Convert values as read from map file, to kG and cm units.	2*UnitConv.	2*E
TITL	Title. Start with “FLIP” to get field map X-flipped.		A80
IX	Number of longitudinal nodes of the map	≤ 400	I
FNAME [, SUM] [1, 2]	File name		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	Unused	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* (*e.g.*, solenoid.map) contains the field data. These must be formatted according to the following *FORTRAN* sequence :

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
DO 1 I = 1, IX
 IF (BINARY) THEN
      READ(NL) X(I), BX(I)
 ELSE
      READ(NL,*) X(I), BX(I)
 ENDIF
  1    CONTINUE
```

where $X(I)$ and $BX(I)$ are the longitudinal coordinate and field component at node (I) of the mesh. Binary file names must begin with *FNAME* 'B_' or 'b_'. 'Binary' will then automatically be set to '.TRUE.'.

[2] Sumperimposing (summing up) field maps is possible. Up to 4 at most.
To do so, pile up file names with 'SUM' following each name but the last one.
In the following example *e.g.*, 3 field maps are read and summed up˜:

```
myMapFile1  SUM
myMapFile2  SUM
myMapFile3
```

(all maps must all have their mesh defined in identical coordinate frame).

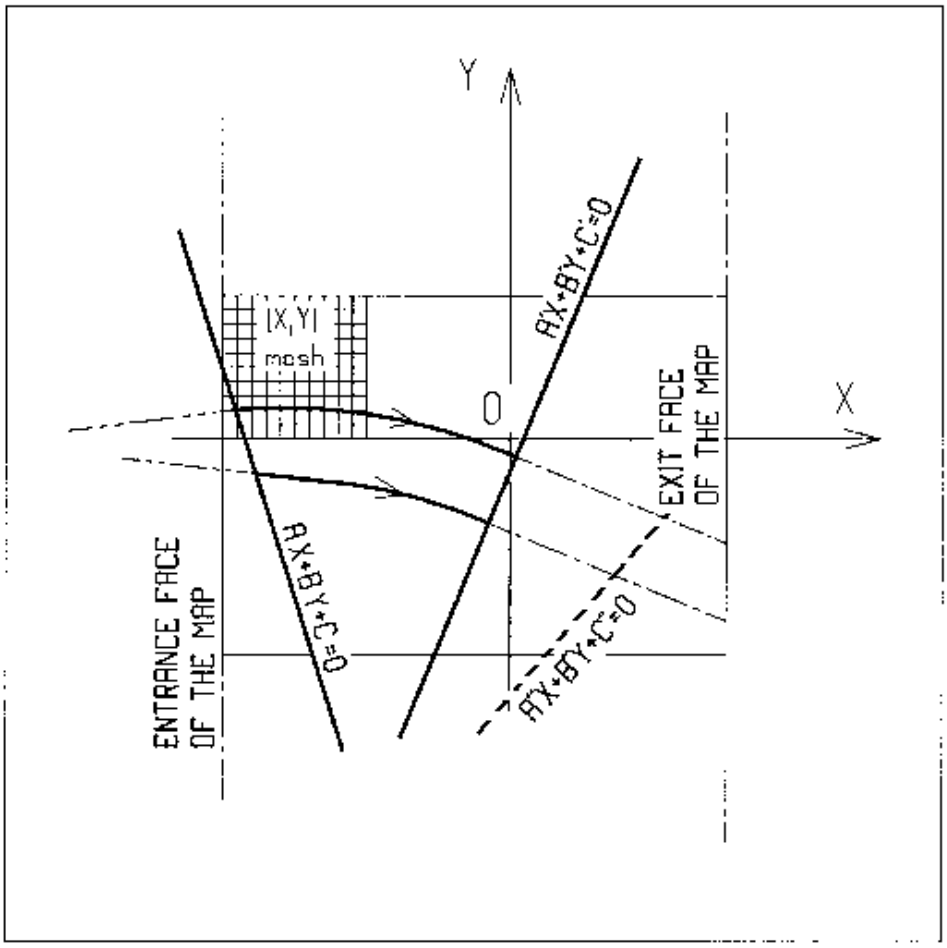
CARTEMES	**2-D Cartesian uniform mesh magnetic field map** mid-plane symmetry is assumed		
IC, *IL*	$IC = 1, 2$: print the map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
BNORM, XN,YN	Field and X-,Y-coordinate normalization coeffs. Convert values as read from map file, to kG and cm units.	3*UnitConv.	3*E
TITL	Title. Start with "FLIP" to get field map X-flipped.		A80
IX, JY	Number of longitudinal (IX) and transverse (JY) nodes of the map	≤ 400, ≤ 200	2*I
FNAME [1]	File name		A80
ID, A, B, C [, A', B', C', A'', B'',etc., if $ID \geq 2$]	Integration boundary. Normally $ID = 0$. $ID = -1$: integration in the map begins at entrance boundary defined by $AX + BY + C = 0$. $ID = 1$: integration in the map is terminated at exit boundary defined by $AX + BY + C = 0$. $ID \geq 2$: entrance (A, B, C) and up to $ID - 1$ exit $(A', B', C', A'', B'', etc.)$ boundaries	≥ -1,2*no dim., cm [,2*no dim., cm, etc.]	I, 3*E [3*E,etc.]
IORDRE	Degree of interpolation polynomial (see *DIPOLE-M*)	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[2] *FNAME* (*e.g.*, spes2.map) contains the field data. These must be formatted according to the following *FORTRAN* sequence :

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
IF (BINARY) THEN
  READ(NL) (Y(J), J=1, JY)
ELSE
        READ(NL,100) (Y(J), J=1, JY)
      ENDIF
100     FORMAT(10 F8.2)
      DO 1 I=1,IX
  IF (BINARY) THEN
    READ(NL) X(I), (BMES(I,J), J=1, JY)
 ELSE
   READ(NL,101) X(I), (BMES(I,J), J=1, JY)
101  FORMAT(10 F8.2)
      ENDIF
 1     CONTINUE
```

where $X(I)$ and $Y(J)$ are the longitudinal and transverse coordinates and *BMES* is the Z field component at a node (I, J) of the mesh. For binary files, *FNAME* must begin with 'B_' or 'b_'.

'Binary' will then automatically be set to '.TRUE.'



OXY is the coordinate system of the mesh. Integration zone limits may be defined, using $ID \neq 0$: particle coordinates are extrapolated linearly from the entrance face of the map, into the plane $A'X + B'Y + C' = 0$; after ray-tracing inside the map and terminating on the integration boundary $AX + BY + C = 0$, coordinates are extrapolated linearly to the exit face of the map.

CAVITE [1]	**Accelerating cavity** $\Delta W = qV\sin(2\pi hf\Delta t + \varphi_s)$ and other voltage and frequency laws.		
IOPT[.i] [,PRINT] [,Facility]	Option. $i = 1$ or 'PRINT' causes info output into `zgoubi.CAVITE.Out` 0-7 'Facility' : 'CornellSynch' (used with *IOPT=11*) or 'eRHIC_RCS' (*IOPT=11*)		I[.I] [A, A]
If IOPT=0	Element inactive		
X, X	Unused.		
X, X	Unused.		
If IOPT=1 [2]	f_{RF} follows the timing law given by *SCALING*		
$\mathcal{L}$, h	Reference closed orbit length ; harmonic number.	m, no dim.	2*E
$\hat{V}$, X	R.F. peak voltage ; unused.	V, unused	2*E
If IOPT=2	f_{RF} follows $\Delta W_s = q\hat{V}sin\phi_s$. Handles SR loss, stationnary bucket . Handles double RF.		
$\mathcal{L}$, h or $\{h_1, h_2\}$	Reference closed orbit length ; harmonic number(s).	m, no dim.	2*E
$\hat{V}$ or $\{\hat{V}_1, \hat{V}_2\}$, ϕ_s	R.F. peak voltage(s) ; synchronous phase.	V, rad	2*E
If IOPT=3	No synchrotron motion : $\Delta W = q\hat{V}sin\phi_s$		
X, X	Unused ; unused.	2*unused.	2*E
$\hat{V}$, ϕ_s	R.F. peak voltage ; synchronous phase.	V, rad	2*E
If IOPT=6	Read RF frequency and/or phase law from external file, "zgoubi.freqLaw.In".		
$\mathcal{L}$, E_k	Orbit length and kinetic energy at start of acceleration.	m, MeV	2*E
$\hat{V}$, Φ_s	R.F. peak voltage ; synchronous phase.	V, rad	2*E
If IOPT=7	Quasi- or isochronous acceleration.		
X, f_{RF}	Unused ; RF frequency.	- , Hz	2*E
$\hat{V}$, Φ_s	R.F. peak voltage ; synchronous phase.	V, rad	2*E
If IOPT=10	Chambers matrix method.		
L, f_{RF}, *[, ID]*	Cavity length ; RF frequency ; damp option, ID=0 (default : just a drift), or $ID = \pm 1$ ($dE/E \ll 1$ hyp. and, resp^{ly} damped or determinant=1), or $ID = \pm 2$ (no hyp. on dE/E and, resp^{ly} damped or determinant=1).	m, Hz	2*E [,I]
$\hat{V}$, Φ_s, *IOPT*	R.F. peak voltage ; synchronous phase ; matrix options.	V, rad, -2-$+2$	2*E, I
If IOPT=11	Acceleration in presence of SR loss.		
$\mathcal{L}$, E_k	Orbit length and kinetic energy at start of acceleration.	m, MeV	2*E
$\hat{V}$, Φ_s, U_0	R.F. peak voltage ; synchronous phase ; SR loss at first pass [3].	V, rad	2*E

`Routines concerned, mostly :` rcavit.f (read input), cavite.f (cavity models).

[1] Use *PARTICUL* to declare mass and charge.
[2] For ramping the R.F. frequency following $B\rho(t)$, use *SCALING*, with family *CAVITE*.
[3] SR loss at subsequent turns is actualized using $U \propto E^4$.

CHAMBR	**Long transverse aperture limitation** [1]		
IA	0 : element inactive 1 : (re)definition of the aperture 2 : stop testing and reset counters, print information on stopped particles.	0-2	I
IFORM[.J], $C1$, $C2$, $C3$, $C4$	*IFORM* = 1 : rectangular aperture ; *IFORM* = 2 : elliptical aperture. *J* = 0, default : opening is [2] $\pm YL = \pm C1$, $\pm ZL = \pm C2$, centered at $YC = C3$, $ZC = C4$. *J* = 1 : opening is [2], in Y : $[C1, C2]$, in Z : $[C3, C4]$	1-2[.0-1]	I[.I], 4*E

[1] Any particle out of limits is stopped.

[2] When used with an optical element defined in polar coordinates (*e.g.*, *DIPOLE*) YL is the radius and YC stands for the reference radius (normally, $YC \simeq RM$).

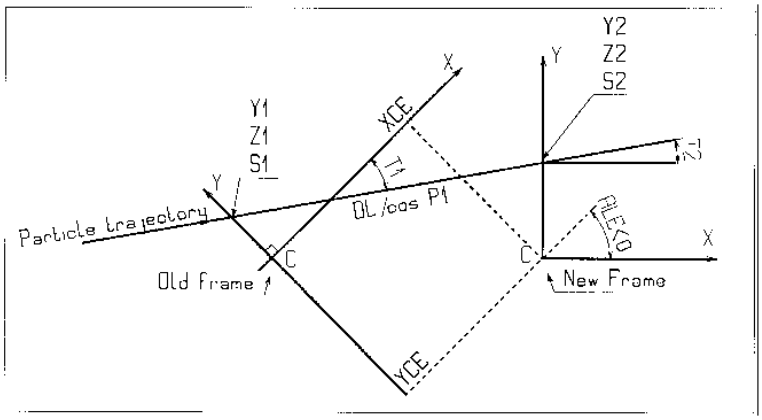
CHANGREF **Transformation to a new reference frame**

“Old Style” (Figure below) :

XCE, YCE, ALE	Longitudinal and transverse shifts, followed by *Z*-axis rotation	2*cm, deg	3*E

“New Style” (example below). In an arbitrary order, up to 9 occurrences of :

XS ’val’, YS ’val’, ZS ’val’, *XR ’val’, YR ’val’, ZR ’val’*	cm or deg	up to 9*(A2,E)



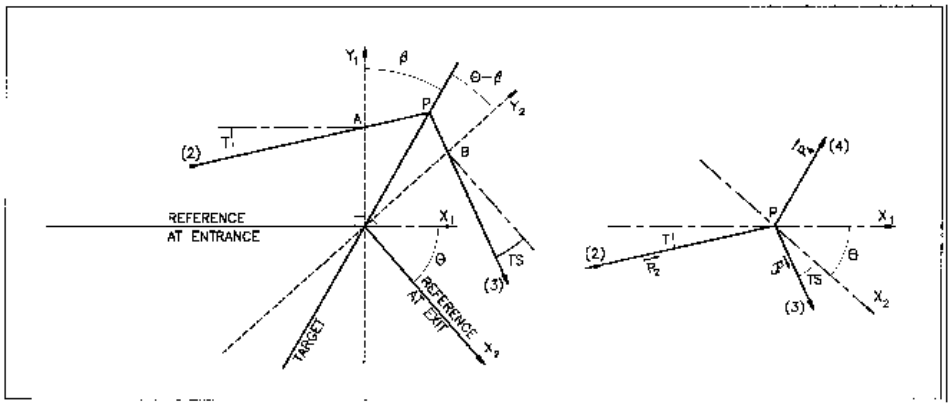
Parameters in the *CHANGREF* procedure.

Example :

```
Using CHANGREF "New Style
'OBJET'
51.71103865921708                          Electron, Ekin=15MeV.
2
1 1                                           One particle, with
2. 0.  0.0 0.0 0.0 1. 'R'      Y_0=2 cm, other coordinates zero.
1 1 1 1 1 1 1
'MARKER'    BEG    .plt               -> list  into zgoubi.plt.
'DRIFT'                                          10 cm drift.
10.
'CHANGREF'
ZR -6.34165 YS 1.             First half Z-rotate, Next Y-shift.
'MULTIPOL'     Combined function multipole, dipole + quadrupole.
   2                                  -> list  into zgoubi.plt.
5  10. 2.064995867082342  2. 0. 0. 0. 0. 0. 0. 0. 0.
 0 0  5. 1.1  1.00 1.00 1.00 1.00 1.00 1. 1. 1. 1.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
 0 0  5. 1.1  1.00 1.00 1.00 1.00 1.00 1. 1. 1. 1.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0 0 0 0 0 0 0 0 0 0
.1   step size
1  0. 0.  0.
'CHANGREF'
YS -1. ZR -6.341         First Y-shift back, next half Z-rotate.
'DRIFT'                                          10 cm drift.
10.
'FAISCEAU'
'END'
```

CIBLE, TARGET **Generate a secondary beam following target interaction**

M_1, M_2, M_3, Q	Target, incident and scattered particle masses ;	5*$\frac{MeV}{c^2}$, 2*deg	7*E
T_2, θ, β	Q of the reaction ; incident particle kinetic energy ; scattering angle ; angle of the target		
NT, NP	Number of samples in T and P coordinates after *CIBLE*		2*I
TS, PS, DT	Sampling size ; tilt angle	3*mrad	3*E
BORO	New reference rigidity after *CIBLE*	kG.cm	E



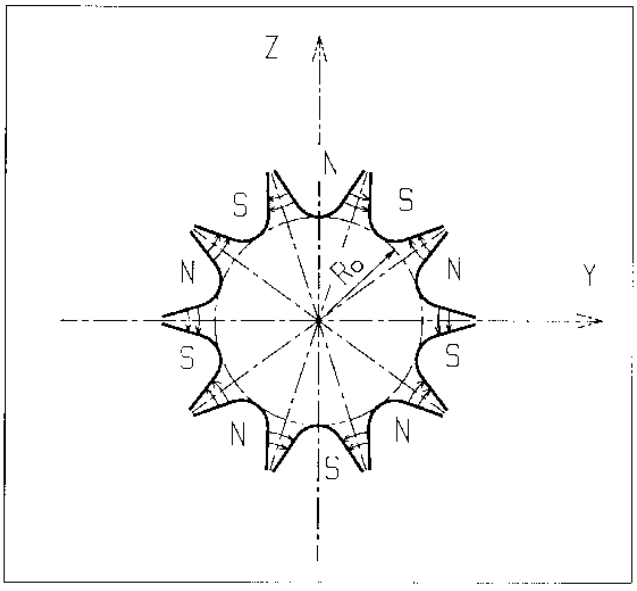
Scheme of the principles of *CIBLE (TARGET)*

A, T = position, angle of incoming particle 2 in the entrance reference frame
P = position of the interaction
B, T = position, angle of the secondary particle in the exit reference frame
θ = angle between entrance and exit frames
β = tilt angle of the target

COLLIMA	**Collimator** [1]		
IA	0 : element inactive 1 : element active 2 : element active and print information on stopped particles	0-2	I
Physical-space collimation			
IFORM[.J], $C1$, $C2$, $C3$, $C4$	*IFORM* = 1 : rectangular aperture ; *IFORM* = 2 : elliptical aperture. *J* = 0, default : opening is $\pm YL = \pm C1$, $\pm ZL = \pm C2$, centered at $YC = C3$, $ZC = C4$. *J* = 1 : opening is, in Y : $[C1, C2]$, in Z : $[C3, C4]$	1-2[.0-1]	I[.I], 4*E
Longitudinal collimation			
IFORM.J, H_{min}, H_{max}, V_{min}, V_{max}	*IFORM* = 6 or 7 for horizontal variable resp[ly] S or Time, J=1 or 2 for vertical variable resp[ly] 1+dp/p, kinetic-E (MeV) ; horizontal and vertical limits	2*cm or 2*s, 2*no.dim or 2*MeV	I, 4*E
Phase-space collimation			
IFORM, α, β, ϵ/π, N_σ	*IFORM* = 11, 14 : horizontal collimation ; horizontal ellipse parameters (unused if 14) [2], emittance, cut-off *IFORM* = 12, 15 : vertical collimation ; vertical ellipse parameters (unused if 15) [2], emittance, cut-off *IFORM* = 13, 16 : longitudinal collimation ; *to be implemented*	11-16, no.dim, 2*m, no.dim	I, 4*E

[1] Any particle out of limits is stopped.
[2] The rejection boundary is the *rms* ellipse matched to the particle distribution.

DECAPOLE	**Decapole magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length ; radius and field at pole tip	2*cm, kG	3*E
X_E, λ_E	Entrance face : Integration zone extent ; fringe field extent ($\lesssim 2R_0$, $\lambda_E = 0$ for sharp edge)	2*cm	2*E
NCE, $C_0 - C_5$	*NCE* = unused $C_0 - C_5$ = Fringe field coefficients such that $G(s) = G_0/(1 + \exp P(s))$, with $G_0 = B_0/R_0^4$ and $P(s) = \sum_{i=0}^{5} C_i(s/\lambda)^i$	unused, 6*no dim.	I, 6*E
X_S, λ_S	Exit face : see entrance face	2*cm	2*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



DIPOLE **Dipole magnet, polar frame**

$$B_Z = \mathcal{F} B_0 \left(1 + N\left(\frac{R-RM}{RM}\right) + B\left(\frac{R-RM}{RM}\right)^2 + G\left(\frac{R-RM}{RM}\right)^3\right)$$

		dim.	data type	FIT numb.
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I	
AT, RM	Total angular extent of the dipole ; reference radius	deg, cm	2*E	2, 3
ACENT, B_0, N, B, G	Azimuth for positioning of EFBs ; field and field indices	deg., kG, 3*no dim.	5*E	4-8
	ENTRANCE FIELD BOUNDARY			
λ, ξ	Fringe field extent (normally $\simeq$ gap size) ; unused. Exponential type fringe field $F = 1\,/\,(1+\exp(P(s)))$ with $P(s) = C_0 + C_1(\frac{s}{\lambda}) + C_2(\frac{s}{\lambda})^2 + ... + C_5(\frac{s}{\lambda})^5$	cm, unused	2*E	9, 10
NC, $C_0 - C_5$, shift	Unused ; C_0 to C_5 : see above ; EFB shift	0-6, 6*no dim., cm	I, 7*E	11-18
ω^+, θ, R_1, U_1, U_2, R_2	Azimuth of entrance EFB with respect to *ACENT* ; wedge angle of EFB ; radii and linear extents of EFB (use $U_{\frac{1}{2}} = \mp\infty$ when $R_{1,2} = \infty$)	2*deg, 4*cm	6*E	19-24
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)			
λ, ξ	Fringe field parameters	cm, unused	2*E	25, 26
NC, $C_0 - C_5$, shift		0-6, 6*no dim., cm	1, 7*E	27-34
ω^-, θ, R_1, U_1, U_2, R_2	Positioning and shape of the exit EFB	2*deg, 4*cm	6*E	35-40
	LATERAL FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)			
λ, ξ	LATERAL EFB is inhibited if $\xi = 0$	cm, unused	2*E	41-42
NC, $C_0 - C_5$, shift		0-6, 6*no dim., cm	1, 7*E	43-50
ω^-, θ, R_1, U_1, U_2, R_2, $RM3$	Positioning and shape of the EFB	2*deg, 5*cm	7*E	51-57
IORDRE, Resol	Degree of interpolation polynomial : 2 = second degree, 9-point grid 25 = second degree, 25-point grid 4 = fourth degree, 25-point grid ; resolution of flying mesh is *XPAS/Resol*	2, 25 or 4 ; no dim.	I, E	58, 59
XPAS	Integration step	cm	E	60
KPOS	Positioning of the map, normally 2. Two options :	1-2	I	
If KPOS = 2	Positioning as follows :			
RE, TE, RS, TS	Radius and angle of reference, respectively, at entrance and exit of the map.	cm, rad, cm, rad	4*E	
If KPOS = 1	Automatic positioning of the map, by means of			
DP	reference relative momentum	no dim.	E	

DIPOLE-M **Generation of dipole mid-plane 2-D map, polar frame**

$$B_Z = \mathcal{F} B_0 \left(1 + N\left(\frac{R-RM}{RM}\right) + B\left(\frac{R-RM}{RM}\right)^2 + G\left(\frac{R-RM}{RM}\right)^3\right)$$

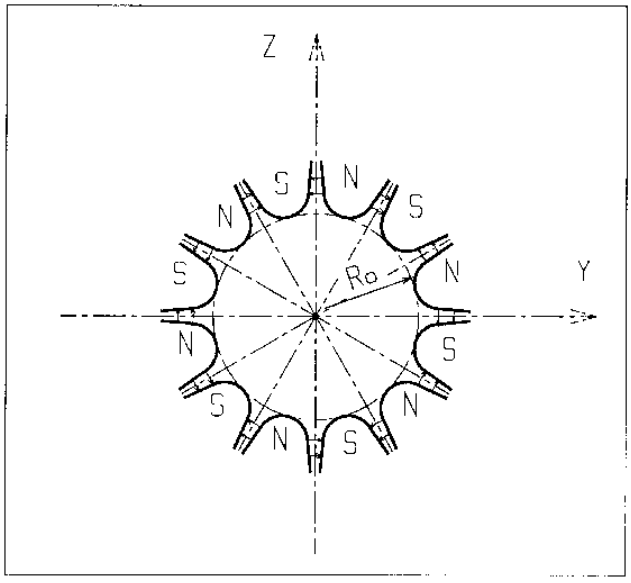
NFACE, IC, IL	Number of field boundaries $IC = 1, 2$: print field map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	2-3; 0-2; 0-2$[\times 10^n]$, 7	3*I
IAMAX, IRMAX	Azimuthal and radial number of nodes of the mesh	≤ 400, ≤ 200	2*I
B_0, N, B, G	Field and field indices	kG, 3*no dim.	4*E
AT, *ACENT*, RM, *RMIN, RMAX*	Mesh parameters : total angle of the map ; azimuth for positioning of EFBs ; reference radius ; minimum and maximum radii	2*deg, 3*cm	5*E
	ENTRANCE FIELD BOUNDARY		
λ, ξ	Fringe field extent (normally $\simeq$ gap size) ; unused. Exponential type fringe field $F = 1\ /\ (1 + \exp(P(s)))$ with $P(s) = C_0 + C_1(\frac{s}{\lambda}) + C_2(\frac{s}{\lambda})^2 + ... + C_5(\frac{s}{\lambda})^5$	cm, unused	2*E
NC, $C_0 - C_5$, shift	Unused ; C_0 to C_5 : see above ; EFB shift	0-6, 6*no dim., cm	I, 7*E
ω^+, θ, R_1, U_1, U_2, R_2	Azimuth of entrance EFB with respect to *ACENT* ; wedge angle of EFB ; radii and linear extents of EFB (use $U_{\frac{1}{2}} = \mp\infty$ when $R_{1,2} = \infty$)	2*deg, 4*cm	6*E
	(Note : $\lambda = 0$, ω^+ = *ACENT* and $\theta = 0$ for sharp edge)		
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
λ, ξ	Fringe field parameters	cm, unused	2*E
NC, $C_0 - C_5$, shift		0-6, 6*nodim., cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2	Positioning and shape of the exit EFB	2*deg, 4*cm	6*E
	(Note : $\lambda = 0$, $\omega^- = -AT +$ *ACENT* and $\theta = 0$ for sharp edge)		
If NFACE = 3	LATERAL FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY) Next 3 records *only* if *NFACE* = 3		
λ, ξ	Fringe field parameters	cm, (cm)	2*E
NC, $C_0 - C_5$, shift		0-6, 6*no dim., cm	I, 7*E
ω^-, θ, R_1, U_1, U_2, R_2, $RM3$	Positioning and shape of the lateral EFB ; RM3 is the radial position on azimuth *ACENT*	2*deg, 5*cm	7*E
NBS	Option index for perturbations to the field map	normally 0	I
If NBS = 0	Normal value. No other record required		
If NBS = -2	The map is modified as follows :		
R_0, $\Delta B/B_0$	B transforms to $B * \left(1 + \frac{\Delta B}{B_0}\frac{R - R_0}{RMAX - RMIN}\right)$	cm, no dim.	2*E
If NBS = -1	the map is modified as follows :		

θ_0, $\Delta B/B_0$	B transforms to $B * \left(1 + \frac{\Delta B}{B_0}\frac{\theta - \theta_0}{AT}\right)$	deg, no dim.	2*E
If NBS ≥ 1	Introduction of NBS shims		
For I = 1, NBS	The following 2 records must be repeated NBS times		
R_1, R_2, θ_1, θ_2, λ	Radial and angular limits of the shim ; λ is unused	2*cm, 2*deg, cm	5*E
γ, α, μ, β	geometrical parameters of the shim	2*deg, 2*no dim.	4*E
IORDRE	Degree of interpolation polynomial : 2 = second degree, 9-point grid 25 = second degree, 25-point grid 4 = fourth degree, 25-point grid	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS	Positioning of the map, normally 2. Two options :	1-2	I
If KPOS = 2 RE, TE, RS, TS	Positioning as follows : Radius and angle of reference, respectively, at entrance and exit of the map.	cm, rad, cm, rad	4*E
If KPOS = 1 DP	Automatic positioning of the map, by means of reference relative momentum	no dim.	E

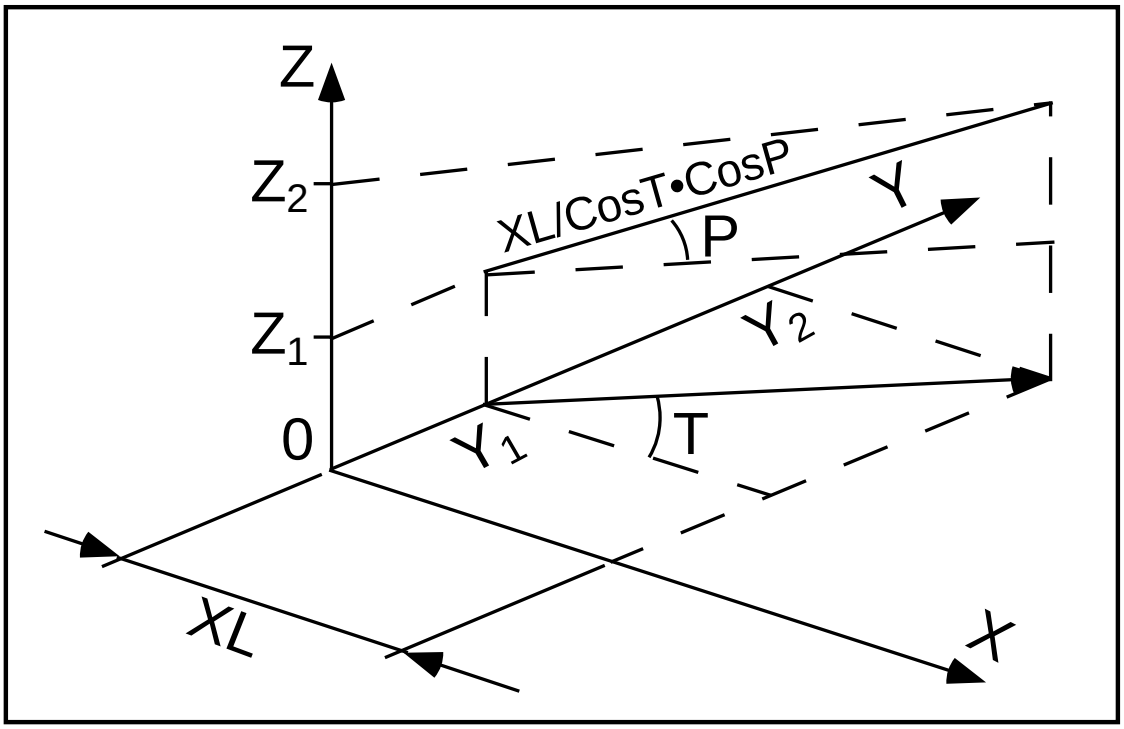
DIPOLES	**Dipole magnet N-tuple, polar frame** (i) $B_Z = \sum_{i=1}^{N} B_{Z0,i}\, \mathcal{F}_i(R,\theta)\ \left(1 + b_{1_i}(R - RM_i)/RM_i + b_{2_i}(R - RM_i)^2/RM_i^2 + ...\right)$ (ii) $B_Z = B_{Z0,i} + \sum_{i=1}^{N} \mathcal{F}_i(R,\theta)\ \left(b_{1_i}(R - RM_i) + b_{2_i}(R - RM_i)^2 + ...\right)$		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
N, AT, RM	Number of magnets in the N-tuple ; total angular extent of the dipole ; reference radius	no dim., deg, cm	I, 2*E
Repeat N times the following sequence			
ACN, δRM [2], B_0, ind, b_i, $(i = 1, ind)$	Positioning of EFBs : azimuth [1], $RM_i = RM + \delta RM$; field ; number of, and field coefficients.	deg., cm, kG, $(ind + 1)$*no dim.	3*E, I, ind*E
	ENTRANCE FIELD BOUNDARY		
g_0, κ	Fringe field extent ($g = g_0\,(RM/R)^{\kappa}$) (0, 0 for hard-edge) [1] Exponential type fringe field $F = 1\,/\,(1 + \exp(P(s)))$ with $P(s) = C_0 + C_1(\frac{s}{g}) + C_2(\frac{s}{g})^2 + ... + C_5(\frac{s}{g})^5$	cm, no dim.	2*E
NC, $C_0 - C_5$, shift	Unused ; C_0 to C_5 : see above ; EFB shift	0-6, 6*no dim., cm	I, 7*E
ω^+, θ, R_1, U_1, U_2, R_2	Azimuth of entrance EFB with respect to *ACN* ; wedge angle of EFB [1]; radii and linear extents of EFB (use $U_{\frac{1}{2}} = \mp\infty$ when $R_{1,2} = \infty$)	2*deg, 4*cm	6*E
	(Note : $g_0 = 0$, ω^+ = *ACENT*, $\theta = 0$ and KIRD=0 for sharp edge)		
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
g_0, κ		cm, no dim.	2*E
NC, $C_0 - C_5$, shift		$0 - 6$, 6*no dim., cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2		2*deg, 4*cm	6*E
	(Note : $g_0 = 0$, $\omega^- = -AT$+ *ACENT*, $\theta = 0$ and KIRD=0 for sharp edge)		
	LATERAL FIELD BOUNDARY **to be implemented - following data not used**		
g_0, κ		cm, no dim.	2*E
NC, $C_0 - C_5$, shift		0-6, 6*no dim., cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2, $RM3$		2*deg, 5*cm	7*E
End of repeat			
KIRD[.n], Resol	If KIRD=0 : analytical computation of field derivatives ; n=0 : default, B_Z formula (i) above, n=1 : B_Z formula (ii). Resol = 2/4 for 2nd/4th order field derivatives computation If KIRD=2, 25 or 4 : numerical interpolation of field derivatives ; size of flying interpolation mesh is *XPAS/Resol* KIRD=2 or 25 : second degree, 9- or 25-point grid KIRD=4 : fourth degree, 25-point grid	0, 2, 25 or 4 ; no dim.	I, E
XPAS	Integration step	cm	E
KPOS	Positioning of the magnet, normally 2. Two options :	1-2	I
If KPOS = 2	Positioning as follows :		
RE, TE, RS, TS	Radius and angle of reference, respectively, at entrance and exit of the magnet	cm, rad, cm, rad	4*E
If KPOS = 1	Automatic positioning of the magnet, by means of		
DP	reference relative momentum	no dim.	E

[1] Hard-edge EFB model requires $\omega^+ = ACN$ (entrance EFB) or $\omega^- = ACN - AT$ (exit) or and $\theta = 0$ wedge-angle.
[2] Non-zero δRM requires KIRD$= 2, 4$ or 25.

DODECAPO	**Dodecapole magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length ; radius and field at pole tip	2*cm, kG	3*E
X_E, λ_E	Entrance face : Integration zone extent ; fringe field extent ($\lesssim 2R_0$, $\lambda_E = 0$ for sharp edge)	2*cm	2*E
NCE, $C_0 - C_5$	*NCE* = unused $C_0 - C_5$ = Fringe field coefficients such that $G(s) = G_0/(1 + \exp P(s))$, with $G_0 = B_0/R_0^5$ and $P(s) = \sum_{i=0}^{5} C_i(s/\lambda)^i$	unused, 6*no dim.	I, 6*E
X_S, λ_S	Exit face : see entrance face	2*cm	2*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



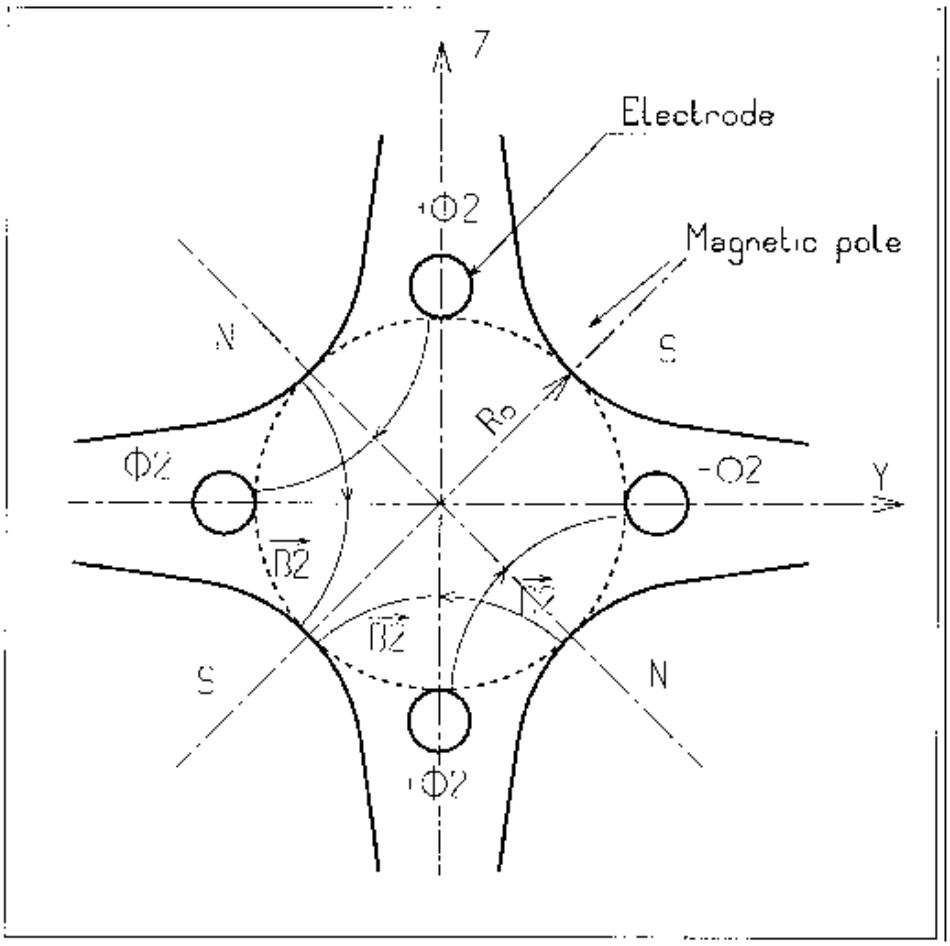
DRIFT, ESL	**Field free drift space**		
XL	length	cm	E



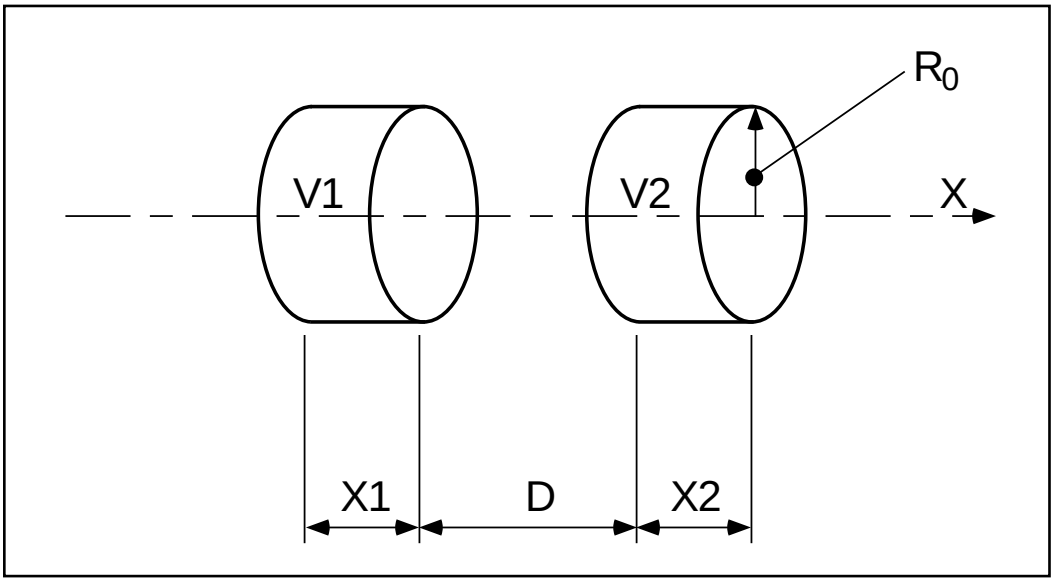
EBMULT [1]	**Electro-magnetic multipole**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
	Electric poles		
XL, R_0, $E1$, $E2$, ..., $E10$	Length of element ; radius at pole tip ; field at pole tip for dipole, quadrupole, ..., 20-pole electric components	2*cm, 10*V/m	12*E
	Entrance face		
X_E, λ_E, E_2, ..., E_{10}	Integration zone ; fringe field extent : dipole fringe field extent = λ_E ; quadrupole fringe field extent = $\lambda_E * E_2$; ... 20-pole fringe field extent = $\lambda_E * E_{10}$ (for any component : sharp edge if field extent is zero)	2*cm, 9*no dim.	11*E
NCE, $C_0 - C_5$	same as *QUADRUPO*	0-6, 6*no dim.	I,6*E
	Exit face		
X_S, λ_S, S_2, ..., S_{10}	Integration zone ; as for entrance	2*cm, 9*no dim.	11*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
$R1$, $R2$, $R3$, ..., $R10$	Skew angles of electric field components	10*rad	10*E
	Magnetic poles		
XL, R_0, $B1$, $B2$, ..., $B10$	Length of element ; radius at pole tip ; field at pole tip for dipole, quadrupole, ..., 20-pole magnetic components	2*cm, 10*kG	12*E
	Entrance face		
X_E, λ_E, E_2, ..., E_{10}	Integration zone ; fringe field extent : dipole fringe field extent = λ_E ; quadrupole fringe field extent = $\lambda_E * E_2$; ... 20-pole fringe field extent = $\lambda_E * E_{10}$ (for any component : sharp edge if field extent is zero)	2*cm, 9*no dim.	11*E
NCE, $C_0 - C_5$	same as *QUADRUPO*	0-6, 6*no dim.	I,6*E

[1] Use *PARTICUL* to declare mass and charge.

	Exit face		
X_S, λ_S, S_2, ..., S_{10}	Integration zone ; as for entrance	2*cm, 9*no dim.	11*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
$R1$, $R2$, $R3$, ..., $R10$	Skew angles of magnetic field components	10*rad	10*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



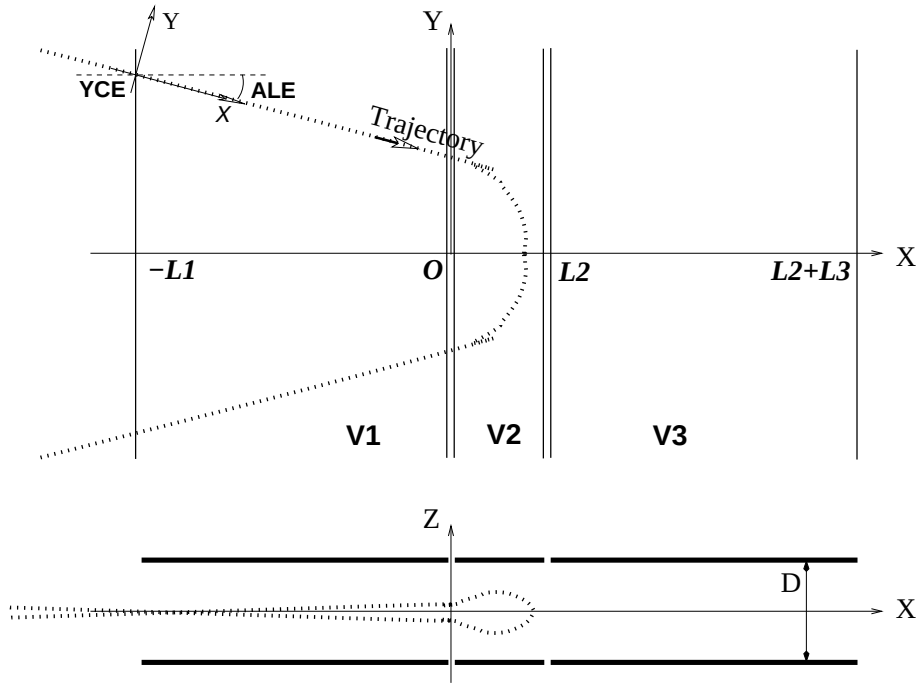
EL2TUB [1]	**Two-tube electrostatic lens**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
X_1, D, X_2, R_0	Length of first tube ; distance between tubes ; length of second tube ; inner radius	3*m	4*E
V_1, V_2	Potentials	2*V	2*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm,	I, 3*E



Two-electrode cylindrical electric lens.

[1] Use *PARTICUL* to declare mass and charge.

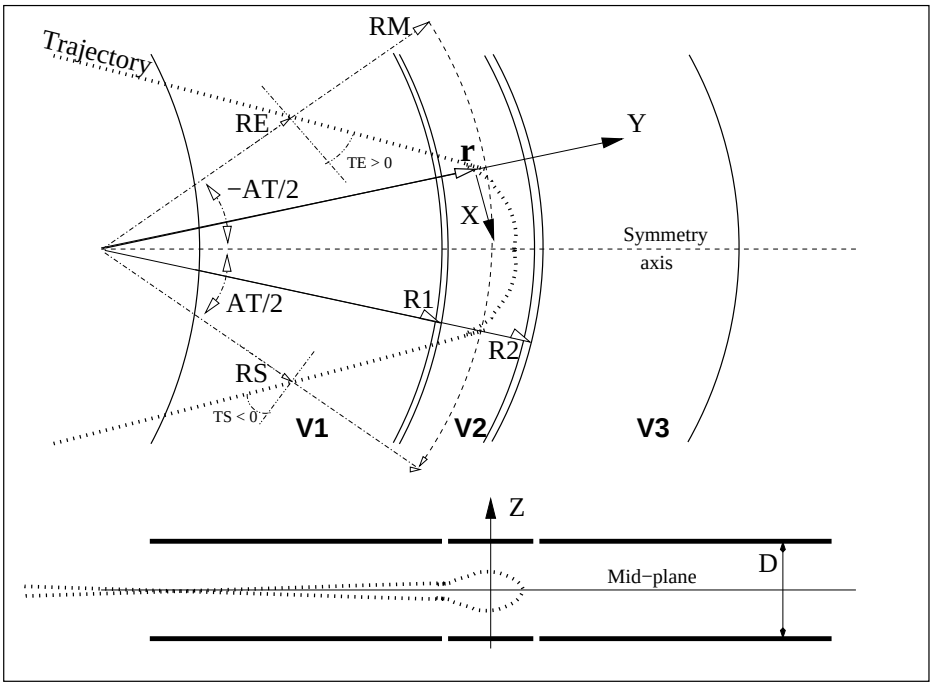
ELMIR	**Electrostatic N-electrode mirror/lens, straight slits**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
N,*L*1, ..., *LN*, *D*, *MT*	Number of electrodes ; electrode lengths ; gap ; mode (11/H-mir, 12/V-mir, 21/V-lens, 22/H-lens)	$2-7$, N*m, m	I, N*E, E, I
*V*1, ..., *VN*	Electrode potentials (normally $V1 = 0$)	N*V	N*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned ; 2 : misaligned ; shifts, tilt ; 3 : automatic positioning, $YCE =$ pitch, $ALE =$ half-deviation	1-2, 2*cm, rad	I, 3*E



Electrostatic N-electrode mirror/lens, straight slits, in the case $N = 3$, in horizontal mirror mode ($MT = 11$).
Possible non-zero entrance quantities YCE, ALE should be specified using *CHANGREF*, or using *KPOS*=3 with *YCE* and *ALE*=half-deviation matched to the reference trajectory.

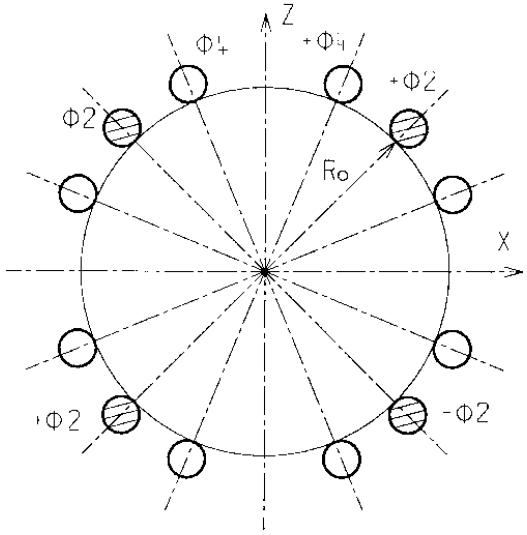
ELMIRC **Electrostatic N-electrode mirror/lens, circular slits**

IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
$R1$, $R2$, AT, D	Radius of first and second slits ; total deviation angle ; gap	4*m 2*m, rad, m	4*E 4*E
$V - VA$, $VB - V$	Potential difference	2*V	2*E
XPAS	Integration step	cm	E
KPOS RE, TE, RS, TS	Normally $KPOS = 2$ for positioning ; Radius and angle at respectively entrance and exit.	1-2 cm, rad, cm, rad	I 4*E



Electrostatic N-electrode mirror/lens, circular slits, in the case $N = 3$, in horizontal mirror mode.

ELMULT [1]	**Electric multipole**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, $E1$, $E2$, ..., $E10$	Length of element ; radius at pole tip ; field at pole tip for dipole, quadrupole, ..., dodecapole components	2*cm, 10*V/m	12*E
	Entrance face		
X_E, λ_E, E_2, ..., E_{10}	Integration zone ; fringe field extent : dipole fringe field extent = λ_E ; quadrupole fringe field extent = $\lambda_E * E_2$; ... 20-pole fringe field extent = $\lambda_E * E_{10}$ (sharp edge if field extent is zero)	2*cm, 9*no dim.	11*E
NCE, $C_0 - C_5$	same as *QUADRUPO*	0-6, 6*no dim.	I, 6*E
	Exit face		
X_S, λ_S, S_2, ..., S_{10}	Integration zone ; as for entrance	2*cm, 9*no dim.	11*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
$R1$, $R2$, $R3$, ..., $R10$	Skew angles of field components	10*rad	10*E
XPAS	Integration step	cm	E
KPOS, *XCE*, *YCE*, *ALE*	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



[1] Use *PARTICUL* to declare mass and charge.

ELREVOL [1]	**1-D uniform mesh electric field map** X-axis cylindrical symmetry is assumed		
IC, *IL*	$IC = 1, 2$: print the map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
ENORM, X-NORM	Field and X-coordinate normalization coeff. Convert values as read from map file, to MV/cm and cm units.	2*UnitConv.	2*E
TITL	Title. Start with "FLIP" to get field map X-flipped.		A80
IX	Number of longitudinal nodes of the map	$\leq$ 400	I
FNAME [2]	File name		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	Unused	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] Use *PARTICUL* to declare mass and charge.

[2] *FNAME* (*e.g.*, e-lens.map) contains the field data. These must be formatted according to the following *FORTRAN* sequence :

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
DO 1 I = 1, IX
  IF (BINARY) THEN
    READ(NL) X(I), EX(I)
  ELSE
    READ(NL,*) X(I), EX(I)
  ENDIF
1   CONTINUE
```

where $X(I)$ and $EX(I)$ are the longitudinal coordinate and field component at node (I) of the mesh. Binary file names *FNAME* must begin with 'B_' or 'b_'. 'Binary' will then automatically be set to '.TRUE.'

EMMA	**2-D Cartesian or cylindrical mesh field map for EMMA FFAG**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
BNORM, XN,YN, ZN	Field and X-,Y-,Z-coordinate normalization coefficients. Convert values as read from map file, to kG and cm or rad.	4*UnitConv.	4*E
TITL	Title. Start with "FLIP" to get field map X-flipped		A80
IX, IY, IZ, *MOD[.i]*	Number of nodes of the mesh in the X, Y and Z directions, $IZ = 1$ for single 2-D map ; *MOD* : operational and map *FORMAT* reading mode [1] *MOD*$\leq$19 : Cartesian mesh ; *MOD*$\geq$20 : cylindrical mesh ; *.i*, optional, tells the reading *FORMAT*, default is '*'.	≤ 400, ≤ 200, 1, ≥ 0[.1-9]	3*I
FNAME [1] ($K = 1, NF$)	Names of the NF files that contain the 2-D maps, ordered from $Z(1)$ to $Z(NF)$. If *MOD*=0 : a single map, superimposition of QF and QD ones, is built for tracking. If *MOD*=1 : a single map, *interpolated* from QF[x_F] and QD[x_D] ones, is built for tracking. If *MOD*=22 : a single map, superimposition of QF and QD ones, is built for tracking. If *MOD*=24 : field at particle is interpolated from a (QF,QD) pair of maps, closest to current (x_F, x_D) value, taken from of set of (QF,QD) pairs registered in FNAME...		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	If $IZ = 1$: as in *CARTEMES* If $IZ \neq 1$: unused	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* normally contains the field map data. If *MOD*=24 *FNAME(K)* contains the names of the QF maps and QD maps, as well as the QF-QD distance attached to each one of these pairs.

ERRORS	**Injecting errors in optical elements** (UNDER DEVELOPMENT)		
ONF, NBR, SEED [, PRINT]	On/off switch (0/1) ; number of error sets to be injected (*i.e.*, as well, number of lines following this one) ; random seed. Occurence of *PRINT* will save error series in zgoubi.ERRORS.out.	,,,[,]	I1, I, I [,A5]

The next line depends on the optical element of concern, and is to be one of the following :
(only limited possibilities at the moment, under development)

MULTIPOL[{LBL1 [,LBL2]}], *N, TYP, AR, UG,* *VC, HW, CUTOFF*	Keyword concerned [optionally, first and/or second label] ; N=1-10 : pole concerned if TYP=BP, otherwise unused ; TYP= BP : field at pole (dipole to 20-pole), or XR : roll-angle ; AR=A or R : error value is absolute or relative (to current one) ; UG=U or G : uniform or Gaussian random law ; VC= central value ; HW= half-width (case UG=U) or sigma (case UG=G) ; cut-off value in units of sigma (unused if UG=U)	,[,],,,,2*kG,	A8 [,A10[,A10]], I, A2, A1, A1, 3*E
TOSCA[{LBL1 [,LBL2]}], *N, TYP, AR, UG,* *VC, HW, CUTOFF*	Keyword concerned [optionally, first and/or second label] ; N is unused ; TYP= BP : field coefficient *BNORM* ; AR, UG, etc. : see abov	,[,],,,,2*kG,	A8 [,A10[,A10]], I, A2, A1, A1, 3*E

• ***Example***

```
 'ERRORS'
1 3 123466                    !   HW        cut-off
MULTIPOL{KCV}  1  BP R U  0.d0    1e-3       0
MULTIPOL{B}    2  XR A U  0.d0    0.349e-3   0
TOSCA          1  BP A G -1.d-3   1.e-5      3
```

In this example the various attributes of the error keyword take the following values and meanings :
◇ 1st line :

- *ONF = 1* : error setting is on ; *NBR=3* : three error setting lines follow ; *seed = 123456*

◇ 2nd line :

- all *MULTIPOL* keywords with first label "*KCV*" in the optical sequence are concerned ; multipole component ip=1 is affected, *i.e.*, dipole component ; "*BP, R, U, 0.d0, 1e-3, 0*" : a random relative defect dB/B (AR=R) concerning the field $B_{\rm ip}$ (TYP=BP) for that component, will be sorted in a uniform (UG=U) distribution centered on $V_{\rm ip} = 0$ (VC=0.d0) with half-width $HW = 10^{-3}$, CUTOFF at n-sigma (n=0 here) is unused (only used for Gaussian distributions, case UG=G) - $B_{\rm ip}$ is changed to $B_{\rm ip} + V_{\rm ip} + dB/B \times B_{\rm ip}$.

◇ 3rd line :

- all *MULTIPOL* keywords with first label "*B*" in the optical sequence are concerned ; multipole component ip=2 is affected, *i.e.*, quadrupole component ; "*XR, A, U, 0.d0, 0.349e-3, 0* : a random absolute defect $dR_{\rm ip}$ (AR=A) of the roll-angle $R_{\rm ip}$ (TYP=XR) for that component, will be sorted in a uniform (UG=U) distribution centered on $V_{\rm ip} = 0$ (VC=0.d0) with half-width $HW = 0.349 \times 10^{-3}$ rad, CUTOFF at n-sigma (n=0 here) is unused (only used for Gaussian distributions, case UG=G) - $R_{\rm ip}$ is changed to $R_{\rm ip} + V_{\rm ip} + dR_{\rm ip}$.

◇ 4th line :

- all *TOSCA* keywords in the optical sequence are concerned ; field map field coefficient *BNORM* (see TOSCA keyword) is affected ; *BP, A, G, -1.d-3, 1.e-5, 3* : a random absolute defect dBN (AR=A) concerning *BNORM* (TYP=BP) will be sorted in a Gaussian (UG=G) distribution centered on $BNC = -10^{-3}$ (VC=-1.d-3) with *rms* value $HW = 10^{-5}$, CUTOFF 3-sigma - *BNORM* is changed to $BNC + dBN$ (not to $BNORM + BNC + dBN$!). (If AR=R instead : random relative defect dBN/BN generated, and BNORM changed to $VC + dBN/BN$).

FAISCEAU **Print particle coordinates**

Print particle coordinates at the location where the
keyword is introduced in the structure.
If frist label is 'FORCE', will force printout when normally
inhibited (*e.g.*, during FIT, or REBELOTE).

FAISCNL **Store particle coordinates in file FNAME**

FNAME[1]	Name of storage file (*e.g.*, zgoubi.fai, or b_zgoubi.fai for binary storage).	A80

FAISTORE **Store coordinates every IP other pass [, at elements with appropriate label]**

FNAME [1] [,*LABEL(s)*]	Name of storage file (*e.g.* zgoubi.fai). Optional : a series of up to 10 label(s), (the first label of element(s) at the exit of which the store will occur) ; wild card accepted, in the form *'*myLabel'* or *'myLabel*'*. If either *FNAME* or first *LABEL* is 'none' then *FAISTORE* is inhibited. Store occurs at all elements if first *LABEL* is 'all' or 'ALL'.	A80, [, 0-10*A10]
IP	Store every *IP* other pass (when using *REBELOTE* with *NPASS* $\geq IP - 1$).	I

[1] Stored data can be read back from *FNAME* using *OBJET*, *KOBJ* = 3.

FFAG	**FFAG magnet, N-tuple** **UNDER DEVELOPMENT** $B_Z = \sum_{i=1}^{N} B_{Z0,i}\, \mathcal{F}_i(R,\theta)\ (R/R_{M,i})^{K_i}$		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
N, AT, RM	Number of dipoles in the FFAG N-tuple ; total angular extent of the dipole ; reference radius	no dim., deg, cm	I, 2*E
Repeat N times the following sequence			
ACN, δRM, B_{Z0}, K	Azimuth for dipole positioning ; $R_{M,i} = RM + \delta RM$; field at $R_{M,i}$; index	deg, cm, kG, no dim.	4*E
	ENTRANCE FIELD BOUNDARY		
g_0, κ	Fringe field extent ($g = g_0\,(RM/R)^{\kappa}$)	cm, no dim.	2*E
NC, $C_0 - C_5$, shift	Unused ; C_0 to C_5 : fringe field coefficients ; EFB shift	0-6, 6*no dim, cm	I, 7*E
ω^+, θ, R_1, U_1, U_2, R_2	Azimuth of entrance EFB with respect to *ACN* ; wedge angle of EFB ; radii and linear extents of EFB (use $U_{\frac{1}{2}} = \mp\infty$ when $R_{1,2} = \infty$)	2*deg, 4*cm	6*E
	(Note : $g_0 = 0$, ω^+ = *ACENT*, $\theta = 0$ and KIRD=0 for sharp edge)		
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
g_0, κ	Fringe field parameters, see above	cm, no dim	2*E
NC, $C_0 - C_5$, shift		0-6, 6*no dim, cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2		2*deg, 4*cm	6*E
	(Note : $g_0 = 0$, $\omega^- = -AT +$ *ACENT*, $\theta = 0$ and KIRD=0 for sharp edge)		
	LATERAL FIELD BOUNDARY **to be implemented - following data not used**		
g_0, κ		cm, no dim	2*E
NC, $C_0 - C_5$, shift		0-6, 6*no dim, cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2		2*deg, 4*cm	6*E
End of repeat			
KIRD, Resol	If KIRD=0 : analytical computation of field derivatives ; Resol = 2/4 for 2nd/4th order field derivatives computation If KIRD $= 2, 4$ or 25 : numerical interpolation of field derivatives ; size of flying interpolation mesh is *XPAS/Resol* KIRD=2 or 25 : second degree, 9- or 25-point grid KIRD=4 : fourth degree, 25-point grid	0, 2, 25 or 4 ; no dim.	I, E
XPAS	Integration step	cm	E
KPOS	Positioning of the magnet, normally 2. Two options :	1-2	I
If KPOS = 2	Positioning as follows :		
RE, TE, RS, TS	Radius and angle of reference, respectively, at entrance and exit of the magnet	cm, rad, cm, rad	4*E
If KPOS = 1	Automatic positioning of the magnet, by means of		
DP	reference relative momentum	no dim.	E

FFAG-SPI	**Spiral FFAG magnet, N-tuple** **UNDER DEVELOPMENT** $B_Z = \sum_{i=1}^{N} B_{Z0,i}\, \mathcal{F}_i(R,\theta)\ (R/R_{M,i})^{K_i}$		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates along trajectories, fields, etc., into zgoubi.res (1) or zgoubi.plt ($2[\times 10^n]$) or zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
N, AT, RM	Number of dipoles in the FFAG N-tuple ; total angular extent of the dipole ; reference radius.	no dim., deg, cm	I, 2*E
Repeat N times the following sequence			
ACN, δRM, B_{Z0}, K	Azimuth for dipole positioning ; $R_{M,i} = RM + \delta RM$; field at $R_{M,i}$; index.	deg, cm, kG, no dim.	4*E
	ENTRANCE FIELD BOUNDARY		
g_0, κ	Fringe field extent ($g = g_0\,(RM/R)^{\kappa}$)	cm, no dim.	2*E
NC, $C_0 - C_5$, shift	Unused ; C_0 to C_5 : fringe field coefficients ; EFB shift	0-6, 6*no dim, cm	I, 7*E
ω^+, ξ, 4 dummies	Azimuth of entrance EFB with respect to *ACN* ; spiral angle ; 4×unused.	2*deg, 4*unused	6*E
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
g_0, κ	Fringe field parameters, see above	cm, no dim	2*E
NC, $C_0 - C_5$, shift		0-6, 6*no dim, cm	1, 7*E
ω^-, ξ, 4 dummies		2*deg, 4*unused	6*E
	LATERAL FIELD BOUNDARY **to be implemented - following data not used**		
g_0, κ		cm, no dim	2*E
NC, $C_0 - C_5$, shift		0-6, 6*no dim, cm	1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2		2*deg, 4*cm	6*E
End of repeat			
Integration boundaries - next line is optional, starting with string `IntLim` :			
`IntLim`, ID, A, B, C [, A', B', C']	Integration boundary. Line has to start with '`IntLim`'. $ID = -1$: integration in the magnet begins at entrance boundary defined by A, B, C. $ID = 1$: integration is terminated at exit boundary defined by A', B', C'. $ID = 2$: both entrance and exit boundaries.	$-1, 1, 2$; deg; cm; deg [; *id.*]	I, 3*E [,3*E]
KIRD, Resol	If KIRD=0 : analytical computation of field derivatives ; Resol = 2/4 for 2nd/4th order field derivatives computation. If KIRD $= 2, 4$ or 25 : numerical interpolation of field derivatives ; size of flying interpolation mesh is *XPAS/Resol.* KIRD=2 or 25 : second degree, 9- or 25-point grid KIRD=4 : fourth degree, 25-point grid	0, 2, 25 or 4 ; no dim.	I, E
XPAS	Integration step	cm	E
KPOS, RE, TE, RS, TS	Positioning of the magnet, has to be 2. As follows : radius and angle of reference, respectively, at entrance and exit of the magnet.	2, 2*(cm, rad)	I, 4*E

FIN, END **End of input data list**

Any information in zgoubi.dat following these keywords will be ignored

FIT, FIT2	**Fitting procedure**		
NV [, nofinal] *[, noSYSout]* *[, save [, FileName]]*	NV : Number of physical parameters to be varied ; 'nofinal' avoids final run (default is : final run performed using fitted values, once fit is done) ; 'noSYSout' inhibits system output of variable and constraint status, except for start and end, and update of penalty value in between ; 'save' saves fit variables when fit is completed, either in 'FileName' if specified, or by default in zgoubi.FITVALS.out.	≤ 20 [, nofinal] save [string]	I [, A7] [, A4 [, A80]]
For I = 1, NV	*repeat NV times the following sequence*		
either :			
IR, IP, XC, DV	Number of the element in the structure ; number of the physical parameter in the element ; coupling switch (off = 0) ; variation range ($\pm$).	$\leq$MXL [1], $\leq$MXD [1], $\pm$ MXL.MXD [2], relative	2*I, 2*E
or :			
IR, IP, XC, $[V_{min}, V_{max}]$	V_{min}, V_{max} : lower and upper limits of the variable.	see footnote[3]	2*I, 3*E
NC [, Penalty [,ITER]] [4]	Number of constraints [, penalty [, max. numb. of iterations]].	≤ 20 [,10^{-n} [,> 0]]	I [, E [, I]]
For I = 1, NC	*repeat NC times the following sequence :*		
IC, I, J, *IR*, V [3], WV, NP $[,\ p_i(i = 1, NP)]$	*IC*, I and J define the type of constraint (see table below) ; IR : number of the element after which the constraint applies ; V : value ; W : weight (the stronger the lower WV) NP : number of parameters ; if $NP \geq 1$, $p_i(i = 1, NP)$: parameter values.	0-5, 3*($>$0), current unit, 2*no dim., curr. units	4*I, 2*E, I, $NP*$E

[1] The values for the maximum number of elements (*i.e.*, keywords in zgoubi.dat) that a sequence in **zgoubi** can contain, MXL, and for the maximum number of parameters under a keyword, MXD, are set in the include file `MXLD.H`.

[2] The coupling input data *XC* is of the form "integer.ijk" with integer$\leq$MXD and i, j, k 1-digit integers and such that ijk$\leq$MXD.

[3] V is in current **zgoubi** units in the case of particle coordinates (*i.e.*, cm, mrad, μs, momentum relative to BORO) and B field (kG). It is in MKSA units (m, rad) in the case of matrix coefficients.

[4] FIT[2] will stop when the sum of the squared residuals gets $<$ *penalty*, or when the maximum allowed number of iterations is reached.

Type of constraint	**Parameters defining the constraints**						**Recommended *[MC]OBJET* ; comments**
	IC	**I**	**J**	**Constraint**	**Additional parameter(s)** *NP*	Param. values, $pr_1 - pr_{NP}$	
Transported σ-matrix $(\sigma(s) = T\sigma(0)\tilde{T})$	0	1 - 6	1 - 6	σ_{IJ} $(\sigma_{11} = \beta_Y, \sigma_{21,\,12} = \alpha_Y$, etc.)			*OBJET/KOBJ=5.1*
Periodic σ-matrix $(\sigma = I\cos\mu + J\sin\mu)$ (N=1-9 for *MATRIX* block 1-9)	0.N $(N \leq 9)$	1 - 6 7 8 9 10	1 - 6 any any any any	σ_{IJ} $(\sigma_{11} = \cos\mu_Y + \alpha_Y \sin\mu_Y$, etc.) $\mu_Y/2\pi$ $\mu_Z/2\pi$ $\cos(\mu_Y)$ $\cos(\mu_Z)$			*OBJET/KOBJ=5.N*
First order transport coeffs.	1	$1-6$ 7 8	$1-6$ i j	Transport coeff. R_{IJ} $i \neq 8$: YY-determinant ; i=8 : YZ-det. $j \neq 7$: ZZ-determinant ; j=7 : ZY-det.			*OBJET/KOBJ=5*
Second order transport coeffs.	2	$1-6$	$11-66$	Transport coeff. $T_{I,j,k}$ $(j = [J/10], k = J - 10[J/10])$			*OBJET/KOBJ=6*
Trajectory	3	$1-$ *IMAX*	$1-7$	$F(J,I)$			*[MC]OBJET*
coordinates		-1	$1-7$	$< F(J,i) >_{i=I1,I2}$	$\begin{cases}0\\2\end{cases}$ $I_1,\quad I_2$		$1 \rightarrow$ *IMAX* $1 \leq I_1 \leq I_2 \leq$ *IMAX*
(I = particle number; J=1-7 for D,Y,T,Z,P,S,time)		-2	$1-7$	$Sup(\lvert F(J,i)\rvert)_{i=1,IMAX}$			
		-3	$1-7$	$Dist\ \ F(J,I)_{i=I1,I2}$	3	I_1, I_2, ΔI	$1 \leq I_1 \leq I_2 \leq$ *IMAX*
		-4	$1-7$	$Dist\,[PU_i, i = 1, N]$	2	$NOEL_A$, $NOEL_B$	PU range
	3.1	$1-$ *IMAX*	$1-7$	$\lvert F(J,I) - FO(J,I)\rvert$			
	3.2	$1-$ *IMAX*	$1-7$	$\lvert F(J,I) + FO(J,I)\rvert$			
	3.4	$1-$ *IMAX*	$1-7$	$\lvert F(J,I) - F(J,K)\rvert$	1	K	$K \leq$ *IMAX*
	3.5	$1-$ *IMAX*	$1-7$	$(F(J,I) - F(J,K))/F(J,K)$	1	K	$K \leq$ *IMAX*
Ellipse parameters	4	$1-6$	$1-6$	σ_{IJ} $(\sigma_{11} = \beta_Y,$ $\sigma_{12} = \sigma_{21} = \alpha_Y$, etc.)			*OBJET/KOBJ=8 ;* *MCOBJET/KOBJ=3*
Number of particles	5	-1	any	$N_{survived}/IMAX$			*OBJET*
		$1-3$	any	$N_{in\ \epsilon_{Y,Z,X}}/N_{survived}$	1	$\epsilon_{Y,Z,X}/\pi$	*MCOBJET*
		$4-6$	any	$N_{in\ best\ \epsilon_{Y,Z,X,rms}}/N_{survived}$			*MCOBJET*
Coordinates & fields, across optical elements (J=1, 2, 3 for $B_{X,\,Y,\,Z}$)	7.1	$1-$ *IMAX*	$1-7$	min. $(pr_1 = 1)$ or max. (2) of $F(J,I)$	1	1-2	*[MC]OBJET*
	7.2	$1-$ *IMAX*	$1-7$	max($F(J,I)$) - min$F(J,I)$)			
	7.3	$1-$ *IMAX*	$1-7$	min$F(J,I)$) + max($F(J,I)$)			
	7.6	$1-$ *IMAX*	$1-3$	min. $(pr_1 = 1)$ or max. (2) value of B_J	1	1-2	
	7.7	$1-$ *IMAX*	$1-3$	max(B_J) - min(B_J)			
	7.8	$1-$ *IMAX*	$1-3$	min(B_J) + max(B_J)			
	7.9	$1-$ *IMAX*	$1-3$	$\int B_J\, ds$			
Spin	10	$1-$ *IMAX*	$1-4$	$S_{X,Y,Z}(I)$, $\lvert\vec{S}(I)\rvert$			*SPNTRK+*
	10.1	$1-$ *IMAX*	$1-3$	$\lvert S_{X,Y,Z}(I) - SO_{X,Y,Z}(I)\rvert$			*[MC]OBJET*
	10.2	$1-$ *IDMAX*	any	spin rotation angle (rad)			*OBJET/KOBJ=2*
	10.3	$1-$ *IDMAX*	1-3	rotation axis, X-, Y- or Z-component			*OBJET/KOBJ=2*

IMAX: total number of particles tracked.
F(J,I) : particle coordinate array ; J : coordinate (1-7 for respectively D, Y, T, Z, P, S, time, resp. units : none, cm, mrad, cm, mrad, cm, μs) ; I : particle number.
FO(J,I) : initial particle coordinates.
S(J,I) : spin coordinate array ; J : coordinate (1, 2, 3 for respectively X, Y, Z) ; I : particle number.
SO(J,I) : initial spin coordinates.

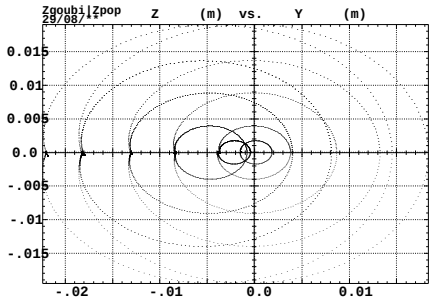
• **Combining FIT and REBELOTE** : An example

In the example below, *FIT* requests that (i) the particle trajectory with initial coordinates defined by *OBJET* have identical horizontal coordinates at both ends of the snake (this is achieved by varying Y_0, T_0 in *OBJET*), and that (ii) the trajectory across the snake - an helix - be Y-centered along the snake axis.

The way this works :

FIT is executed a first time for $B\rho_{ref} = BORO = 7205.1782956$ kG.cm, namely the execution loops between *OBJET* and *FIT* until the constraints are best fulfilled. Once this is completed, the execution pointer then goes to the next instruction in the data list, namely '*SPNPRT*', this is discussed below, and then points to *REBELOTE*, which will have the effect of sending it back to the begining of zgoubi.dat data list. However, prior to that, the "**1**" flag, fourth data in first row in *REBELOTE*, requests a change of, next line, parameter number **35** in element "**OBJET**" (that is, the relative regidity of the particle, D). The list of values follows, namely, {1.3872739973 2.1368296674 4.8261190694 11.015241321}. The change occurs **4** times, according to *NRBLT=4*, the first data in *REBELOTE* data list.
Note that, (i) spin data so computed (spin vector components, precession, etc) are stored/stacked in zgoubi.SPNPRT.Out by placing the keyword '*SPNPRT*' with label "*PRINT*", between *FIT* and *REBELOTE* ; (ii) current updated FIT variables are saved in zgoubi.FITVALS.out using the '*save*' command in *FIT*, and further stacked in zgoubi.FITVALS.out_cat for each *REBELOTE* case, using *SYSTEM*.

```
Centering 5 helical orbits in the AGS warm helical snake 3-D OPERA map.
 'OBJET'            ! This data list may be copy-pasted and run, as is.
7.2051782956D3      ! Reference rigidity of the problem
2
1 1                 ! A single particle. Initial coordinates :
-2.2  0. 0. 0. 0.  1. 'o'           ! Yo, To, Zo, Po, So, p/po
1
 'PARTICUL'         ! proton data are necessary for spin tracking
938.27203 1.602176487E-19 1.7928474 0 0     ! M, Q, G factor
 'SPNTRK'
4.1                 ! Initial spin is positionned vertical
0. 0. 1.

 'FAISCEAU'
 'SPNPRT'

 'TOSCA'
0  20
1.e1   100. 100. 100.
HEADER_0 wsnake
801 29 29 12.1      ! The map is a 801x29x29 node 3-D mesh
warmSnake.map       ! AGS warm snake 3-D OPERA map
0 0 0 0
2
.1
2  0.  .0  0.  0.

 'FAISCEAU'
```



This figure shows how, using *REBELOTE*, the helical trajectories at five different momenta are moved by the *FIT* procedure, one after the other, from initial off-centered position to final centering on the snake axis (at Y=Z=0).

```
 'FIT'
2   save            ! Two variables. Save FIT variables (in zgoubi.FITVALS.out).
1 30 0  [-3,3]          ! Vary initial coordinate Y_0 (horiz. position)
1 31 0  [-3,3]          ! Vary initial coordinate T_0 (horiz. angle)
3    1E-2           ! Three constraints (penalty 2E-3 requested) :
3.1 1 2 6 0. 1. 0       ! Y_0=Y after at exit of the magnet
3.1 1 3 6 0. 1. 0       ! T_0=T after at exit of the magnet
7.3 1 2 6 0. 1. 0       ! Y_min+Y_max=0 inside the OPERA field map

 'SPNPRT'  PRINT    ! Stack spin data (in zgoubi.SPNPRT.Out).
 'SYSTEM'           ! Save zgoubi.FITVALS.out data following from successive REBELOTE.
1
cat zgoubi.FITVALS.out >> zgoubi.FITVALS.out_cat

 'REBELOTE'         ! Will loop on re-doing the FIT for 4 additional particle rigidities
4  0.1  0 1
1                   ! List of 4 successive values of D in OBJET follows
OBJET  35  1.3872739973  2.1368296674  4.8261190694  11.015241321

 'SYSTEM'           ! Save a copy of zgoubi.FITVALS.out_cat and of zgoubi.SPNPRT.Out_cat
2
\cp zgoubi.FITVALS.out_cat zgoubi.FITVALS.out_cat_copy
\cp zgoubi.SPNPRT.Out      zgoubi.SPNPRT.Out_copy
 'END'
```

• **FIT[2] options** : An example

FIT[2] options, “nofinal”, “save FileName”, “penalty” and “ITER”, are all specified in the example below.

```
An example with FIT options
'GETFITVAL'
zgoubi.FITVALS.In
 'MCOBJET'1
57.36635309d3         ! reference rigidity
3.1
1
2 2 2 2 1 1
0. 0. 0. 0. 0.   4.61943880E-01
 2.703371  1.859656  1.48563786487e-99   9
-2.592712  2.421706  1.48563786487e-99   9
0. 1.  0.     9
12345 23456 34567
 'PARTICUL'                                                                    2
0.51099892 1.60217653e-19 1.15965218076e-3 0.0 0.0
'PICKUPS'
  2
 #StartRing MULT BD2S #EndRing
...
 'DRIFT'    DRIF      HD                                                      7497
14.38218115
 'MARKER'  #SSMid                                                             7498
 'MARKER'  #EndRing                                                           7500
 'FIT2'
4 save zgoubiFIT.result nofinal  ! Variables saved in zgoubiFIT.result when FIT2 done. No final run.
!!! 4 save nofinal  ! Would cause variables to be saved in default file zgoubi.FITVALS.out instead.
2 40 0 [-1.e-2,1.e-2]              ! The four variables are Yo, To, Zo, Po in MCOBJET
2 41 0 [-1.e-2,1.e-2]
2 42 0 [-1.e-2,1.e-2]
2 43 0 [-1.e-2,1.e-2]
4   1e-10   90                ! penalty=1e-10. Maximum allowed calls to zgoubi is 90.
3.1 1 2 7500 0. 1. 0          ! The four constraints are : initial Y, T, Z, P of
3.1 1 3 7500 0. 1. 0          ! particle #1 equal to its final Y, T, Z, P.
3.1 1 4 7500 0. 1. 0
3.1 1 5 7500 0. 1. 0
```

FOCALE	**Particle coordinates and horizontal beam size at distance *XL***		
XL	Distance from the location of the keyword	cm	E

FOCALEZ	**Particle coordinates and vertical beam size at distance *XL***		
XL	Distance from the location of the keyword	cm	E

GASCAT	**Gas scattering**		
KGA	Off/On switch	0, 1	I
AI, DEN	Atomic number ; density		2*E

GETFITVAL **Get values of *variables* as saved from former FIT[2] run**

FNAME	Name of storage file. Zgoubi will proceed silently if FNAME=’none’ or FNAME=’NONE’, or if the file is not foumd.	A

GOTO	**Branching statement** **Under development**		
OPTION	Can be ’PASS#’ or ’GOBACK’	String	A
If OPTION = PASS# *LABEL_1, ..., LABEL_N*	One line needed : List of labels (expected is label #1 that follows a keyword) to go to, pass after pass (passes are incremented, from 1 to N, by *REBELOTE*..	N strings	N*A

• Using **GOTO** : An example is given page 243

HISTO **1-D histogram**

J, $X_{\min}$, $X_{\max}$, *NBK*, *NH*	J = type of coordinate to be histogrammed ; the following are available : • current coordinates : $1(D)$, $2(Y)$, $3(T)$, $4(Z)$, $5(P)$, $6(S)$, • initial coordinates : $11(D_0)$, $12(Y_0)$, $13(T_0)$, $14(Z_0)$, $15(P_0)$, $16(S_0)$, • spin : $21(S_X)$, $22(S_Y)$, $23(S_Z)$, $24(< S >)$; $X_{\min}$, $X_{\max}$ = limits of the histogram, in units of the coordinate of concern ; *NBK* = number of channels ; *NH* = number of the histogram (for independence of histograms of the same coordinate)	1-24, 2* current units, < 120, 1-5	I, 2*E, 2*I
NBL, KAR, NORM, TYP	Number of lines (= vertical amplitude) ; alphanumeric character ; normalization if *NORM* = 1, otherwise *NORM* = 0 ; *TYP* = ‘P’ : primary particles are histogrammed, or ‘S’ : secondary, or Q : all particles - for use with *MCDESINT*	normally 10-40, char., 1-2, P-S-Q	I, A1, I, A1

IMAGE **Localization and size of horizontal waist**

IMAGES **Localization and size of horizontal waists**

For each momentum group, as classified by means of *OBJET*, *KOBJ* = 1, 2 or 4

IMAGESZ **Localization and size of vertical waists**

For each momentum group, as classified by means of *OBJET*, *KOBJ* = 1, 2 or 4

IMAGEZ **Localization and size of vertical waist**

INCLUDE	**File include statement**		
NBF	Number of files to be included.	-	I
NBF following lines (one file name per line) :			
FNAME [, [LBL_1A [, LBL_2A]]: [LBL_1B [, LBL_2B]]	Name of input zgoubi.dat-style file to be included. *LBL_A* : entrance branching tag, *LBL_B* : exit branching tag.		A80 [,[A10], [A10]]

• Using **INCLUDE** : An example.

In this example of an energy-recovery electron recirculator [69], the linac, spreader and combiner data lists are given in separate files, declared as *INCLUDE*s. Using *INCLUDE* has a series of merits :

- it shortens the parent input data file, so making the optical structure more apparent,

- it allows picking optical spreader, combiner and ring sequences from files that may actually be set to - for instance - compute optical parameters ; in that case, LBL_A and LBL_B are used for this targetted picking : they define the ends of the sequence to be *INCLUDE*'d, ignoring the upper and lower parts in the *INCLUDE*'d file (which may be specific, for instance, to MATRIX computations).

Note the following : (i) juggling between linac, spreaders, combiners and recirculating ring, is handled using *GOTO* statement, (ii) the use of *REBELOTE* in this ERL context is discussed in an example page 271.

```
Cornell CBETA prototype (of an eRHIC) FFAG arc ERL
 'MCOBJET'
19.9411300960454126   ! T.m    E_tot=6MeV
5.01
.01 .01 .01 .01 .01 .01
0. 0. 0. 0. 0. 1.
-0.8475 1.5739 -0.8481 1.59 0. 1. 7.49E-03  4.12E-03 0. 0.
 'PARTICUL'
0.51099892 1.60217653e-19 1.15965218076e-3 0.0 0.0

 'ERRORS'
0 1 123466                   dB(kG)
MULTIPOL{}  1  BP A U 0.d0   0.0  9999

 'SCALING'
1  2
MULTIPOL
2
1.  1.
1 9999
MULTIPOL SA1.1 SB1.1
2
1.  1.
1 9999

'OPTIONS'
1  1
WRITE ON

!////// Racetrack sequence starts here, at linac entrance
! Linac
'INCLUDE'
1
./LA/LA_obj5.inc[LA.MAR.BEG\1:LA.MAR.END\1]

! Goto splitter line
 'GOTO'
PASS#
SA1 SA2 SA3 SA4 SA3 SA2 SA1
 'MARKER' RT_SA

! Goto FFAG return loop : arc+(DS+straight+DS)+arc
 'GOTO'
PASS#
FA FA FA FA FA FA FA
 'MARKER' RT_FA
 'GOTO'
PASS#
 X X X X X X X
 'MARKER' RT_X
 'GOTO'
PASS#
FB FB FB FB FB FB FB
 'MARKER' RT_FB

! Goto combiner line
 'GOTO'
PASS#
SB1 SB2 SB3 SB4 SB3 SB2 SB1
 'MARKER' RT_SB

 'MATRIX'
1 0

 'MARKER'  rebelote
 'REBELOTE'  ! send zgoubi pointer back to
7  0.1  99   ! the top (and linac entrance), 7 times

 'MARKER'  stop
 'STOP'
!/////////////////////
!// job done, here //
!/////////////////////
```

```
! Splitter lines
'MARKER'  SA1
'INCLUDE'
1
./SA/SA1.inc[LA.MATCH1:S1.MERGE.MAR.END\1]
 'GOTO'
GOBACK
'MARKER'  SA2
'INCLUDE'
1
./SA/SA2.inc[LA.DEMER.MAR.BEG\2:S1.MERGE.MAR.END\2]
 'GOTO'
GOBACK
'MARKER'  SA3
'INCLUDE'
1
./SA/SA3.inc[SA3#S:SA3#E]
 'GOTO'
GOBACK
'MARKER'  SA4
'INCLUDE'
1
./SA/SA4.inc[SA4#S:SA4#E]
 'GOTO'
GOBACK

'MARKER'  FA
'INCLUDE'
1
./FA/FA.inc[FA#S:FA#E]
 'GOTO'
GOBACK

'MARKER'  X
'INCLUDE'
1
./X./X.inc[X#S:X#E]
 'GOTO'
GOBACK

'MARKER'  FB
'INCLUDE'
1
./FB/FB.inc[FB#S:FB#E]
 'GOTO'
GOBACK

! Combiner lines
'MARKER'  SB1
'INCLUDE'
1
./SB/SB1.inc[SB1#S:SB1#E]
 'GOTO'
GOBACK
'MARKER'  SB2
'INCLUDE'
1
./SB/SB2.inc[SB2#S:SB2#E]
 'GOTO'
GOBACK
'MARKER'  SB3
'INCLUDE'
1
./SB/SB3.inc[SB3#S:SB3#E]
 'GOTO'
GOBACK
'MARKER'  SB4
'INCLUDE'
1
./SB/SB4.inc[SB4#S:SB4#E]
 'GOTO'
GOBACK

'MATRIX'
1  0
'END'
```

MAP2D	**2-D Cartesian uniform mesh field map - arbitrary magnetic field**		
IC, *IL*	$IC = 1, 2$: print the field map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
BNORM, XN,YN	Field and X-,Y-coordinate normalization coeffs. Convert values as read from map file, to kG and cm units.	3*UnitConv.	3*E
TITL	Title. Start with “FLIP” to get field map X-flipped.		A80
IX, JY	Number of longitudinal and horizontal-transverse nodes of the mesh (the Z elevation is arbitrary)	≤ 400, ≤ 200	2*I
FNAME [1]	File name		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	Degree of polynomial interpolation, 2nd or 4th order.	2, 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* (*e.g.,* magnet.map) contains the field map data.
These must be formatted according to the following *FORTRAN* read sequence (normally compatible with *TOSCA* code *OUTPUTS* - details and possible updates are to be found in the source file `'fmapw.f'`):

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD')
DO 1 J = 1, JY
 DO 1 I = 1, IX
   IF (BINARY) THEN
         READ(NL) Y(J), Z(1), X(I), BY(I,J), BZ(I,J), BX(I,J)
   ELSE
         READ(NL,100) Y(J), Z(1), X(I), BY(I,J), BZ(I,J), BX(I,J)
100        FORMAT (1X, 6E11.4)
       ENDIF
1    CONTINUE
```

where $X(I)$, $Y(J)$ are the longitudinal, horizontal coordinates in the at nodes (I, J) of the mesh, \$Z(1)\$ is the vertical elevation of the map, and BX, BY, BZ are the components of the field.

For binary files, *FNAME* must begin with 'B_' or 'b_'~; a logical flag 'Binary' will then automatically be set to '.TRUE.'

MAP2D-E	**2-D Cartesian uniform mesh field map - arbitrary electric field**		
IC, *IL*	$IC = 1, 2$: print the field map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
ENORM, X-,Y-NORM	Field and X-,Y-coordinate normalization coeffs. Convert values as read from map file, to MV/cm and cm units.	3*UnitConv.	3*E
TITL	Title. Start with "FLIP" to get field map X-flipped.		A80
IX, JY	Number of longitudinal and horizontal-transverse nodes of the mesh (the Z elevation is arbitrary)	≤ 400, ≤ 200	2*I
FNAME [1]	File name		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	Degree of polynomial interpolation, 2nd or 4th order.	2, 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* (*e.g.*, ''mirror.map'') contains the field map data.
These must be formatted according to the following *FORTRAN* read sequence - details and possible updates are to be found in the source file `'fmapw.f'` :

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD')
DO 1 J = 1, JY
  DO 1 I = 1, IX
    IF (BINARY) THEN
          READ(NL) Y(J), Z(1), X(I), EY(I,J), EZ(I,J), EX(I,J)
    ELSE
          READ(NL,100) Y(J), Z(1), X(I), EY(I,J), EZ(I,J), EX(I,J)
100         FORMAT (1X, 6E11.4)
        ENDIF
1     CONTINUE
```

where $X(I)$, $Y(J)$ are the longitudinal, horizontal coordinates in the at nodes (I, J) of the mesh, $Z(1)$ is the vertical elevation of the map, and EX, EY, EZ are the components of the field.

For binary files, *FNAME* must begin with 'B_' or 'b_'˜; a logical flag 'Binary' will then automatically be set to '.TRUE.'

MARKER **Marker**

Just a marker. No data
'.plt' as a second *LABEL* will cause storage of current coordinates into zgoubi.plt

MATRIX **Calculation of transfer coefficients, periodic parameters**

IORD, IFOC Options : 0-2, 0-1 or > 10 2*I [,A]
[, PRINT] [, coupled]

IORD = 0 : Same effect as *FAISCEAU*
IORD = 1, using *OBJET*, *KOBJ* = 5[.N] : 1st order transfer parameters,
- if *KOBJ* = 5.1 : beam matrix, phase advance, otherwise
- 1st order data (*cf.* IFOC value) for N different cases of optical reference.

IORD = 2, using *OBJET*, *KOBJ* = 6 : 1st order transfer matrix $[R_{ij}]$,
2nd order array $[T_{i\,jk}]$ and a few higher order transfer coefficients ;
some periodic parameters in addition if *IFOC* > 10 ;
IORD = 3, using *OBJET*, *KOBJ* = 6.1 : 1st order transfer matrix $[R_{ij}]$,
2nd and 3rd order arrays $[T_{i\,jk}]$, $[T_{i\,jkl}]$.

IFOC = 0 : matrix at actual location [1] ;
IFOC = 1 : matrix at the closest first order horizontal focus [1] ;
IFOC = 10 + NPER : same as *IFOC* = 0, and also calculates 1 or N sets
(case respectively of *KOBJ* = 5 or *KOBJ* = 5.N, $2 \leq N \leq 99$) of Twiss
parameters, tune numbers, etc., assuming zgoubi.dat file describes
one period of a NPER-period structure [1].

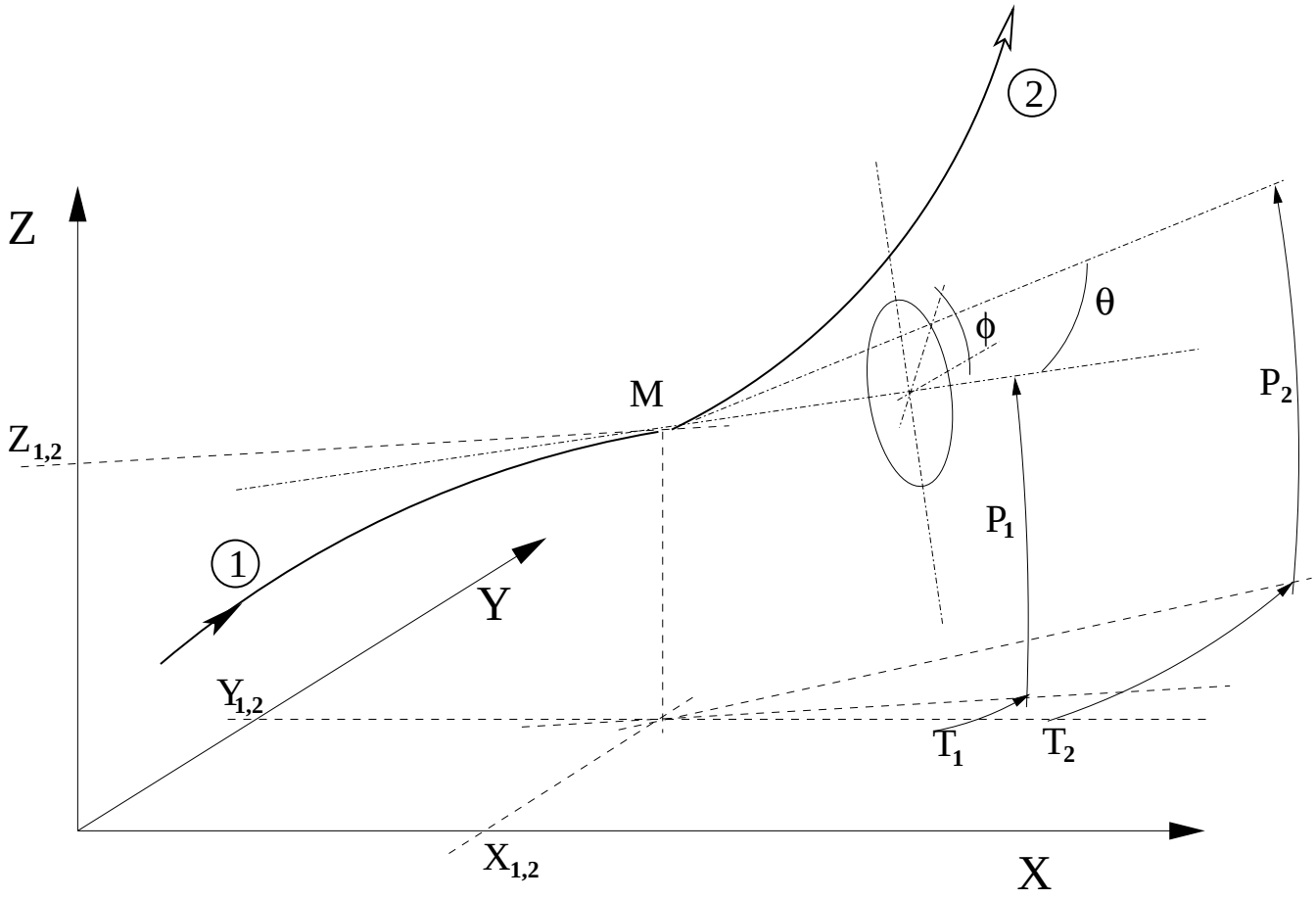
Occurence of "*PRINT*" will cause printout to zgoubi.MATRIX.out file.

Occurence of "*coupled*", in the periodic case (*IFOC* = 10 + NPER, above),
will cause use of coupled formalism.

[1] The reference in computing transport coefficients and other periodic parameters is particle # 1 of the ray-traced set. If N 11-particle sets are ray-traced in order to get (in a single zgoubi run) transport coeeficients for N different cases of reference orbit (using *OBJET*, *KOBJ* = 5.N, $2 \leq N \leq 99$), the reference particle for each 11-particle set is respectively particle #1, 12, 23, ... 1+(N-1)$\times$11.

MCDESINT [1] **Monte-Carlo simulation of in-flight decay**
M1 → M2 + M3

[INFO,] [2] $M2$, $M3$, $\tau 2$ [3]	[Switch,] ; masses of the two decay products; COM lifetime of particle 2	[-,] 2*MeV/c^2, s	[A4,] 3*E
$I1$, $I2$, $I3$	Seeds for random number generators	3*$\simeq 10^6$	3*I



Particle 1 decays into 2 and 3 ; **zgoubi** then calculates trajectory of 2, while 3 is discarded. θ and ϕ are the scattering angles of particle 2 relative to the direction of the incoming particle 1. They transform to T_2 and P_2 in Zgoubi frame.

[1] *MCDESINT* must be preceded by *PARTICUL*, for the definition of the mass and lifetime of the incoming particle M1.
[2] Presence of 'INFO' will cause more info on decay kinematics parameters to be printed zgoubi.res at each decay.
[3] $\tau 2$ can be left blank, in which case the lifetime of particle 2 is set to zero (it decays immediately, which from a practical point of view means that it is not tracked).

MCOBJET	**Monte-Carlo generation of a 6-D object**		
BORO	Reference rigidity	kG.cm	E
KOBJ	Type of support of the random distribution *KOBJ*= 1 : window *KOBJ*= 2 : grid *KOBJ*= 3 : phase-space ellipses	1-3	I
IMAX	Number of particles to be generated	$\leq 10^4$	
KY, KT, KZ, KP, KX, KD [1]	Type of probability density	6*(1-3)	6*I
Y_0, T_0, Z_0, P_0, X_0, D_0	Mean value of coordinates ($D_0 = B\rho/BORO$)	m, rad, m, rad, m, no dim.	6*E
If KOBJ = 1	**In a window**		
δY, δT, δZ, δP, δX, δD	Distribution widths, depending on KY, KT etc. [1]	m, rad, m, rad, m, no dim.	6*E
$N_{\delta Y}$, $N_{\delta T}$, $N_{\delta Z}$, $N_{\delta P}$, $N_{\delta X}$, $N_{\delta D}$	Sorting cut-offs (used only for Gaussian density)	units of σ_Y, σ_T, etc.	6*E
N_0, C_0, C_1, C_2, C_3	Parameters involved in calculation of P(D)	no dim.	5*E
$IR1$, $IR2$, $IR3$	Random sequence seeds	3*$\simeq 10^6$	3*I
If KOBJ = 2	**On a grid**		
IY, IT, IZ, IP, IX, ID	Number of bars of the grid		6*I
PY, PT, PZ, PP, PX, PD	Distances between bars	m, rad, m rad, m, no dim.	6*E
δY, δT, δZ, δP, δX, δD	Width of the bars ($\pm$) if uniform, Sigma value if Gaussian distribution	*ibidem*	6*E
$N_{\delta Y}$, $N_{\delta T}$, $N_{\delta Z}$, $N_{\delta P}$, $N_{\delta X}$, $N_{\delta D}$	Sorting cut-offs (used only for Gaussian density)	units of σ_Y, σ_T,etc.	6*E
N_0, C_0, C_1, C_2, C_3	Parameters involved in calculation of $P(D)$	no dim.	5*E
$IR1$, $IR2$, $IR3$	Random sequence seeds	3*$\simeq 10^6$	3*I

[1] Let $x = Y, T, Z, P$ or X. KY, KT, KZ, KP and KX can take the values

1 : uniform, $p(x) = 1/2\delta x$ if $-\delta x \leq x \leq \delta x$

2 : Gaussian, $p(x) = \exp(-x^2/2\delta x^2)/\delta x\sqrt{2\pi}$

3 : parabolic, $p(x) = 3(1 - x^2/\delta x^2)/4\delta x$ if $-\delta x \leq x \leq \delta x$

KD can take the values

1 : uniform, $p(D) = 1/2\delta D$ if $-\delta D \leq x \leq \delta D$

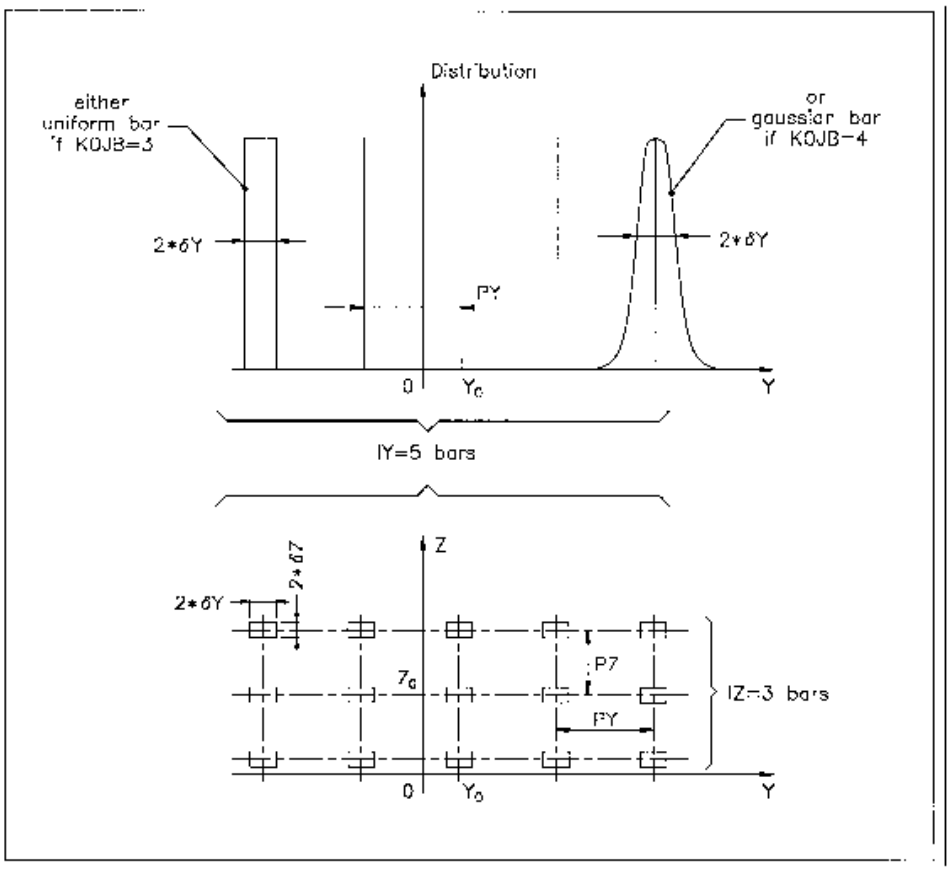
2 : exponential, $p(D) =$ No $\exp(C_0 + C_1 l + C_2 l^2 + C_3 l^3)$ if $-\delta D \leq x \leq \delta D$

3 : kinematic, $D = \delta D * T$

If KOBJ = 3	**On a phase-space ellipse [1]. Four lines follow :**		
α_Y, β_Y, ε_Y/π, $N_{\sigma_{\epsilon_Y}}$ [, $N'_{\sigma_{\epsilon_Y}}$ if $N_{\sigma_{\epsilon_Y}} < 0$] [2] [, D_Y, D_T]	Ellipse parameters and emittance, Y-T phase-space ; cut-off ; dispersion and derivative	no dim., m/rad, m, units of $\sigma(\varepsilon_Y)$; [, m, rad]	4*E [,E] [,2*E]
α_Z, β_Z, ε_Z/π, $N_{\sigma_{\epsilon_Z}}$ [, $N'_{\sigma_{\epsilon_Z}}$ if $N_{\sigma_{\epsilon_Z}} < 0$] [2] [, D_Z, D_P]	Ellipse parameters and emittance, Z-P phase-space ; cut-off ; dispersion and derivative	no dim., m/rad, m, units of $\sigma(\varepsilon_Z)$ [, m, rad]	4*E [,E] [,2*E]
α_X, β_X, ε_X/π, $N_{\sigma_{\epsilon_X}}$ [, $N'_{\sigma_{\epsilon_X}}$ if $N_{\sigma_{\epsilon_X}} < 0$] [2]	Ellipse parameters and emittance, X-D phase-space ; cut-off	no dim., m/rad, m, units of $\sigma(\varepsilon_X)$	4*E [,E]
$IR1$, $IR2$, $IR3$	Random sequence seeds	3*$\simeq 10^6$	3*I

[1] Similar possibilities, non-random, are offered with *OBJET*, KOBJ=8 (p. 254)

[2] Works with Gaussian density type only : sorting is within the ellipse frontier ($\frac{1+\sigma_Y^2}{\beta_Y^2}Y^2 + 2\alpha_Y YT + \beta_Y T^2 = \frac{\varepsilon_Y}{\pi}$) if $N_{\sigma_{\epsilon_Y}} > 0$, or, if $N_{\sigma_{\epsilon_Y}} < 0$ sorting is within the annular region [$|N_{\sigma_{\epsilon_Y}}|, N'_{\sigma_{\epsilon_Y}}$] of that ellipse.



A scheme of input parameters to *MCOBJET* when *KOBJ*= 2.
Top : Possible distributions of the Y coordinate
Bottom : A 2-D grid in (Y, Z) space.

MULTIPOL **Magnetic Multipole**

		dim.	data type	FIT numb.
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I	
XL, R_0, $B1$, $B2$, ..., $B10$,	Length of element ; radius at pole tip ; field at pole tip for dipole, quadrupole, ..., dodecapole components	2*cm,10*kG	12*E	
	Entrance face			
X_E, λ_E, E_2, ..., E_{10}	Integration zone ; fringe field extent : dipole fringe field extent = λ_E ; quadrupole fringe field extent = $\lambda_E * E_2$; ... 20-pole fringe field extent = $\lambda_E * E_{10}$ (sharp edge if field extent is zero)	2*cm,9*no dim.	11*E	
NCE, $C_0 - C_5$	same as *QUADRUPO*	0-6, 6*no dim.	I, 6*E	
	Exit face			
X_S, λ_S, S_2, ..., S_{10}	Integration zone ; as for entrance	2*cm, 9*no dim.	11*E	
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E	
$R1$, $R2$, $R3$, ..., $R10$	Skew angles of field components	10*rad	10*E	
XPAS	Integration step	cm	E	62
If KPOS = 1-3	Positioning as follows :			
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned ; *KPOS*=2 : misaligned ; shifts, tilt ; *KPOS*=3 : effective only if $B1 \neq 0$: entrance and exit frames are tilted *wrt.* the magnet by an angle of • either ALE if ALE≠0 • or 2 Arcsin$(B1\,XL\,/\,2BORO)$ if ALE=0	1-3, 2*cm, rad	I, 3*E	
If KPOS = 4				
KPOS, XS, YS, ZR, ZS, YR	X-, Y-shift, Z-rotation, Z-shift, Y-rotation Automatic ALE positioning as *KPOS* = 3, and in addition possible X- or Y- or Z-misalignment or Z-, Y-rotation. (under development).	4, cm, 2*(cm, rad)	I, 5*E	

OBJET	**Generation of an object**		
BORO	Reference rigidity	kG.cm	E
KOBJ[.K2]	Option index [.More options]	1-6	I
If KOBJ = 1[.1]	[Non-] Symmetric object		
IY, IT, IZ, IP, IS, ID	Ray-Tracing assumes mid-plane symmetry. Generated points : $YR \pm IY * dY$, $TR \pm IT * dT$, $ZR \pm IZ * dZ$, $PR \pm IP * dP$ [$ZR + IZ * dZ$, $PR + IP * dP$ if KOBJ = 1.1], $SR \pm IS * dS$ and $DR \pm ID * dD$ coordinates ($IY \leq 20$,...,$ID \leq 20$)	IY*IT*IZ*IP*IS*ID $\leq 10^4$	6*1
dY, dT, dZ, dP, dS, dD	Sampling size in Y, T, Z, P, S and in relative momentum ($dD = \delta B\rho/BORO$)	2(cm,mrad), cm, no dim.	6*E
YR, TR, ZR, PR, SR, DR	Reference trajectory ($DR = B\rho/BORO$)	2(cm,mrad), cm, no dim.	6*E
If KOBJ = 2[.1]	All the initial coordinates must be entered explicitly		
IMAX, IDMAX	total number of particles ; number of distinct momenta (if $IDMAX > 1$, group particles of same momentum)	$IMAX \leq 10^4$	2*I
For I = 1, IMAX	Repeat *IMAX* times the following line		
Y, T, Z, P, S, D, LET	Coordinates and tagging of the *IMAX* particles ; *If KOBJ = 2.1* input units are different :	2(cm,mrad), cm, no dim., 2(m,rad), m, no dim.,	6*E, A1
IEX($I = 1$, *IMAX*)	*IMAX* times 1 or -9. If $IEX(I) = 1$ trajectory I is ray-traced, it is not if $IEX(I) = -9$.	1 or -9	*IMAX*I
If KOBJ=3[.N, N=0 - 3]	Reads coordinates from a storage file N=0 (default) : [b_]zgoubi.fai style data file FORMAT N=1 : read FORMAT is ''READ(NL,*) Y,T,Z,P,S,DP'' N=2 : read FORMAT is ''READ(NL,*) X,Y,Z,PX,PY,PZ'' N=3 : read FORMAT is ''READ(NL,*) DP,Y,T,Z,P,S,TIME,MASS,CHARGE''		
IT1, IT2, ITStep	Read particles numbered IT1 to IT2, step ITStep (For more than 10^4 particles stored in *FNAME*, use '*REBELOTE*')	≥ 1, $\geq IT1$, ≥ 1	3*I
IP1, IP2, IPStep	Read particles that belong in pass numbered IP1 to IP2, step IPStep	≥ 1, $\geq IP1$, ≥ 1	3*I
YF, TF, ZF, PF, SF, DF, TF, TAG	Scaling factor. TAG-ing letter : no effect if *TAG*='*', otherwise only particles with TAG≡LET are retained.	7*no.dim, char.	7*E, A1
YR, TR, ZR, PR, SR, DR, TR	Reference. Given the previous line of data, all coordinate C (=Y, T...) is transformed to C*CF+CR	2(cm, mrad), cm, no dim.,μs	7*E
InitC	0 : set new $\vec{R}_0 = \vec{R}_0$ as read, new $\vec{R} = \vec{R}$ as read ; 1 : set new $\vec{R}_0 = \vec{R}$ as read, new $\vec{R} = \vec{R}$ as read ; 2 : save $\vec{R}$ as read in new $\vec{R}_0$, set new $\vec{R} = \vec{R}_0$ as read.	0-1	I
FNAME	File name (*e.g.*, [b_]zgoubi.fai) (N in KOBJ=3.N determines storage FORMAT)		A80

If KOBJ = 5[.N, $N \geq 1$]	**Generation of 11 particles, or 11*N if** $N \geq 2$ (for use with *MATRIX*, $IORD = 1$)		
dY, dT, dZ, dP, dS, dD	Sampling size in Y, T, Z, P, S (unused) and D	2(cm,mrad), cm, no dim.	6*E
YR, TR, ZR, PR, SR, DR	Reference trajectory (*DR* $= B\rho/$*BORO* ; *SR* is not sused)	2(cm,mrad), cm, no dim.	6*E
◇ *If KOBJ = 5.1* $\alpha_Y, \beta_Y, \alpha_Z, \beta_Z, \alpha_S, \beta_S,$ D_Y, D'_Y, D_Z, D'_Z	One additional data line : Initial beam ellipse parameters [1]	 2(no dim.,m), ?, ?, 2(m,rad)	 6*E, 4*E
◇ *If KOBJ = 5.N (*$N \geq 2$*)* *YR, TR, ZR, PR, SR, DR*	N additional data lines [2] Reference trajectory # i (*DR* $= B\rho/$*BORO* ; *SR* is not sused)	2(cm,mrad), cm, no dim.	6*E
If KOBJ = 6	**Generation of 61 particles** (for use with *MATRIX*, $IORD = 2$)		
dY, dT, dZ, dP, dS, dD	Sampling size in Y, T, Z, P, S (unused) and D	2(cm,mrad), cm, no dim.	6*E
YR, TR, ZR, PR, SR, DR	Reference trajectory ; *DR* $= B\rho/$*BORO*	2(cm,mrad), cm, no dim.	6*E
If KOBJ = 7	**Object with kinematics**		
IY, IT, IZ, IP, IS, ID	Number of points in $\pm Y$, $\pm T$,$\pm Z$, $\pm P$, $\pm S$; ID is not used	*IY*IT*IZ*IP*IS*$\leq 10^4$	6*I
dY, dT, dZ, dP, dS, dD	Sampling size in Y, T, Z, P and S ; *dD* = kinematic coefficient, such that $D(T) = DR + dD * T$	2(cm,mrad), cm, mrad^{-1}	6*E
YR, TR, ZR, PR, SR, DR	Reference ($DR = B\rho/$*BORO*)	2(cm,mrad), cm, no dim.	6*E
If KOBJ = 8	**Generation of phase-space coordinates on ellipses** [3]		
IY, IZ, IS	Number of samples in each 2-D phase-space ; if zero the central value (below) is assigned	$0 \leq IY, IZ, IS \leq IMAX$, $1 \leq IY * IZ * IS \leq IMAX$	3*I
Y_0, T_0, Z_0, P_0, S_0, D_0	Central values ($D_0 = B\rho/$*BORO*)	2(m, rad), m, no dim.	6*E
α_Y, β_Y, ε_Y/π	ellipse parameters and emittances	no dim., m, m	3*E
α_Z, β_Z, ε_Z/π		no dim., m, m	3*E
α_S, β_S, ε_S/π		no dim., m, m	3*E

[1] They can be transported by using MATRIX.
[2] The maximum value for N, normally MXREF=999, is hard-coded in obj5.f.
[3] Similar possibilities, random, are offered with *MCOBJET*, KOBJ=3 (p. 250).

• **OBJET, *KOBJ*=3** recovering from a crash : An example

The job below, a 9 million turn tracking in RHIC, crashed at turn # 2,217,299. It can be read from the storage file b_zgoubi.fai (or infered as well from the *OBJET* and *CAVITE* data), that the reference (synchronous, theoretical) rigidity at the previously saved turn (turn# 2,217,298, the closest integer multiple of 67) was $B\rho_{\rm ref} = 272.519214209$ T.m (magnet scaling coefficient under *SCALING*, right column below), which is $B\rho_{\rm ref}/BORO = 3.111758007$ (as used in *SCALING*, CAVITE data, right column), with $BORO = 87.5772517$ T.m the initial rigidity (25.33 GeV), top of left column.
The right column shows how these informations are transposed to the new tracking run based on *OBJET, KOBJ=3*, which allows resuming tracking from that last saved turn #2,217,298.

This 9,000,000-turn job crashed at turn number 2,217,299. Tracking starts from clock'6 IP in RHIC, acceleration (keyword *CAVITE*) at a rate of $50 \times \cos(2.1618)$ kV per turn.
Particle coordinates are stored turn-by-turn at SNK1 (keyword *FAISTORE*), location of the first helical dipole.

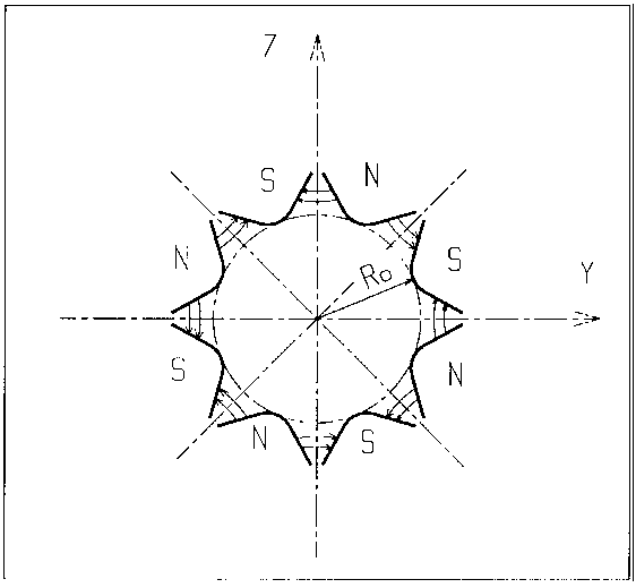
```
'OBJET'
87.5772517e3
2              ! Declare 8 particles, all
8 1            ! on a 21mu_m, norm. vertical invariant.
0.     0.    2.58575455E-01  -1.73487148E-01  0. 1. 'o'
0.     0.    1.82840458E-01   1.21643409E-01  0. 1. 'o'
0.     0.    1.58331802E-17   3.45516907E-01  0. 1. 'o'
0.     0.   -1.82840458E-01   3.66991287E-01  0. 1. 'o'
0.     0.   -2.58575455E-01   1.73487148E-01  0. 1. 'o'
0.     0.   -1.82840458E-01  -1.21643409E-01  0. 1. 'o'
0.     0.   -4.74995405E-17  -3.45516907E-01  0. 1. 'o'
0.     0.    1.82840458E-01  -3.66991287E-01  0. 1. 'o'
1 -1 -1 -1 -1 -1 -1 -1  ! on/off switches. Particle 1 selected.
'PARTICUL'         ! Particle data needed for spin tracking.
9.382720300E+02 1.602176487E-19 1.792847400  0. 0.
'SPNTRK'
3                  ! All initial spins vertical.
'FAISCEAU'
'FAISTORE'         ! Storage location in sequence is at SNK1,
b_zgoubi.fai SNK1  ! the first of the two helicoidal snakes.
67                 ! Storage (at pass 1 and) every 67 passes.
'SCALING'
1  5               ! 5 families of magnets ramped.
BEND               ! Magnet fields strictly follow cavity kick :
-1                 ! (i) NT=-1 can be used, and (ii) CAVITE does
87.5772517         ! not need be declared under SCALING.
1
MULTIPOL HKIC VKIC
-1
0.                 ! Zero field in HKIC and VKIC magnet families.
1
MULTIPOL QUAD
-1
87.5772517
1
MULTIPOL SEXT
-1
87.5772517
1
MULTIPOL MULT
-1
87.5772517
1
 'OPTIONS'
1 1
WRITE OFF
'MARKER'   MARK      RHIC$START
'MARKER'   MARK      CLOCK6down

-------------------------------------------------
----- RHIC ring, from CLOCK6 to CLOCK6 ----------
----- including snake 1 region :
'MARKER' SNK1
'SPINR'    Snake1     OSNKE
1
+45. 180.                ! snakAxis, spin angle
-------------------------------------------------

'MARKER'   MARK      CLOCK6up
'MARKER'   MARK      RHIC$END
'CAVITE'  accelerating cavity
2
3833.8456    120.00 circumf., H
50e3    2.61799387799
'REBELOTE'
8999999   0.4  99    ! Carry on tracking through optical
                     ! sequence 8999999 times.
'END'
```

This run recovers from the crash, starting at SNK1 (a permutation of the previous data file) with coordinates as read from b_zgoubi.fai (storage file as filled by *FAISTORE* during the run that crashed). Magnet and cavity data are updated to their values at turn number 2,217,299.

```
 'OBJET'
87.5772517e3
3
1 8 1                    ! Will look for stored turn in
2217290  2217299 1       ! range (2217298 is concerned).
1. 1. 1. 1. 0. 1. 0. '*' ! Zero factor to 'S' for correct
0. 0. 0. 0. 0. 0. 0.     ! path length/RF phase at CAVITE.
0
b_zgoubi.fai    ! The storage file is used to recover
                ! particle coordinates at pass # 2217299.
'PARTICUL'
9.382720300E+02  1.602176487E-19  1.792847400E+00  0. 0.
'SPNTRK'
3
'FAISCEAU'

'FAISTORE'
b_zgoubi.fai   SNK1
67
'SCALING'
1  6
BEND
-1
272.519214209
1
MULTIPOL HKIC VKIC
-1
0.
1
MULTIPOL QUAD
-1
272.519214209
1
MULTIPOL SEXT
-1
272.519214209
1
MULTIPOL MULT
-1
272.519214209
1
CAVITE
2
1.            3.07555022085
1             6782702

'OPTIONS'
1 1
WRITE OFF

'MARKER' SNK1   ! start point changed to SNK1

-------------------------------------------------
------ RHIC ring, from snake 1 to snake 1 ------
-------------------------------------------------
'SPINR'    Snake1     OSNKE
1
+45. 180.                ! snakAxis, spin angle

'CAVITE'                 ! accelerating cavity,
2                        ! location is arbitrary
3833.8456    120.00
50e3    2.61799387799
'REBELOTE'
6782700   0.4  99
'END'
```

OBJETA	**Object from Monte-Carlo simulation of decay reaction** $M1 + M2 \longrightarrow M3 + M4$ and $M4 \longrightarrow M5 + M6$		
BORO	Reference rigidity	kG.cm	E
IBODY, KOBJ	Body to be tracked : $M3$ (*IBODY*=1), $M5$ (*IBODY*=2) $M6$ (*IBODY*=3) ; type of distribution for Y_0 and Z_0 : uniform (*KOBJ*= 1) or Gaussian (*KOBJ*= 2)	1-3,1-2	2*I
IMAX	Number of particles to be generated (use '*REBELOTE*' for more)	$\leq 10^4$	I
$M_1 - M_6$	Rest masses of the bodies	6*GeV/c^2	6*E
T_1	Kinetic energy of incident body	GeV	E
Y_0, T_0, Z_0, P_0, D_0	Only those particles in the range $Y_0 - \delta Y \leq Y \leq Y_0 + \delta Y$ ..... $D_0 - \delta D \leq D \leq D_0 + \delta D$ will be retained	2(cm,mrad), no dim.	5*E
δY, δT, δZ, δP, δD		2(cm,mrad), no dim.	5*E
XL	Half length of object : $-XL \leq X_0 \leq XL$ (uniform random distribution)	cm	E
$IR1$, $IR2$	Random sequence seeds	2*$\simeq 0^6$	2*I

OCTUPOLE	**Octupole magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length ; radius and field at pole tip of the element	2*cm, kG	3*E
	Entrance face :		
X_E, λ_E	Integration zone ; Fringe field extent ($\lambda_E = 0$ for sharp edge)	2*cm	2*E
NCE, $C_0 - C_5$	*NCE* = unused $C_0 - C_5$ = fringe field coefficients such that : $G(s) = G_0/(1 + \exp\ P(s))$, with $G_0 = B_0/R_0^3$ and $P(s) = \sum_{i=0}^{5} C_i(s/\lambda)^i$	any, 6*no dim.	I, 6*E
	Exit face :		
X_S, λ_S	Parameters for the exit fringe field ; see entrance	2*cm	2*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



Octupole magnet

OPTICS **Write out optical functions. Log to zgoubi.OPTICS.out**

IOPT[.NLBL], label(s) *[, IMP or 'PRINT']* *[, coupled]*	$IOPT = 0/1$: Off/On ; NLBL = number of labels (default is 1) ; 'label' : can be 'all', 'ALL', or a list of *NLBL* first label(s) as appearing at one or more elements in zgoubi.dat sequence ; wild card accepted, in the form *'*myLabel'* or *'myLabel*'*. $IMP = 1$, or presence of 'PRINT', will cause storage of optical functions in zgoubi.OPTICS.out ; optical function computation is in coupled hypothesis if specified, uncoupled otherwise.	0-1[.1-NLBL], - , - , -	I[.I], 1-NLBL*A10, I1 or A5, A7

• ***Example***
The following will cause a print of transported optical functions in zgoubi.res, at all optical elements of the zgoubi.dat sequence, as well as their print out in zgoubi.OPTICS.out :

```
'OPTICS'
 1  all  PRINT
```

OPTIONS **Global or special options**

IOPT, NBOP	$IOPT = 0/1$: Off/On ; 0 (off) will inhibit all *NBOP* subsequent requests. *NBOP* : total number of options.	0-1, ≥ 0	2*I

NBOP lines follow. Possible choices :

'WRITE', OPT	OPT = ON/OFF : allows/inhibites (most) write statements to zgoubi.res	'WRITE', 'ON' or 'OFF'	2*A
'CONSTY', OPT	OPT = ON : forces constant Y and Z across optical elements	'CONSTY', 'ON'/'OFF'	2*A

• **CONSTY** : The example below yields (data saved to zgoubi.plt due to $IL = 2$) 3-D field components (trivially uniform in that particular case, $B_R = B_\theta = 0,\ B_Z = B_0$ at all (R, θ, Z)) on a $n_R \times n_\theta \times n_Z = (41-1)/2 \times (R*AT/XPAS) \times (11-1)/2$ node, uniform 3-D mesh, with $\Delta R = \delta Y$ in *OBJET* $= 1$ cm, $\Delta\theta \equiv XPAS/RM = 10.471975512$ mrd, $\Delta Z = \delta Z$ in *OBJET* $= 0.2$ cm, across a 60 degree hard-edged sector dipole (6 of these make up a uniform field cyclotron).

```
Cyclotron, classical.
Cyclotron, classical.
 'OBJET'
1000.e0
1
41 1  11  1  1  1
1. 0. 0.2 0. 0. 1.
0. 0. 0. 0. 0. 1.

 'OPTIONS'
1 1
CONSTY ON

 'DIPOLES'
2
1  60. 50.
30. 0. 5. 1 0.
0.  0.                                    ! EFB 1  hard-edge
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
30. 0.  1.E6  -1.E6  1.E6  1.E6
0.   0.                                   ! EFB 2
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
-30. 0.  1.E6  -1.E6  1.E6  1.E6
0. 0.                                     ! EFB 3
0  0.      0.      0.      0.      0. 0.  0.
0.  0.   0.    0.    0. 0. 0.
0 2    KIRD, Resol=IDB
0.523598775598
2  00. 0. 00. 0.
 'FAISCEAU'

 'END'
```

ORDRE	**Taylor expansions order**		
IO	Taylor expansions of $\vec{R}$ and $\vec{u}$ up to $\vec{u}^{(IO)}$ (default is $IO = 4$)	2-5	I

PARTICUL **Particle characteristics**

Either :

M, Q, G, τ, X	Mass (or coded, see Note below) ; charge ; gyromagnetic factor ; COM life-time ; unused	MeV/c^2, C, no dim., s, NA	5*E

or :

Type	ELECTRON or MUON+, MUON-, PROTON	A

Note : If M is of the form {`M1 M2`}, then when masses are assigned to particles from a previously defined object, the first half of the particles are given the mass `M1`, and the second half are given the mass `M2`.

If Q is zero, the reference charge is left unchanged.

Only the parameters to be used need their value be specified (for instance M, Q only when electric lenses are used, M, Q, G only when tracking spin) ; unused parameters can be set to zero.

PICKUPS	**Bunch centroid path; orbit** **Summation of particles coordinates at pickups may be printed out in zgoubi.PICKUP.out.** **Pickup contents are zeroed at start of each pass, in multi-turn mode.**		
N	0 : inactive ≥ 1 : number of *LABELs* at which beam centroid is computed	≥ 0	I

A list of N keywords' labels follows

*LABEL*1 [,LABEL2, [...]]	The N label(s) at which beam data are to be computed/recorded. If some "*LABELi*" in this list actually does not appear in zgoubi.dat optical sequence, then it is peacefully ignored ; wild card accepted, in the form *'*myLabel'* or *'myLabel*'*.	N string(s)	N*A10

• ***Example***
A trick :

```
'PICKUPS'
1
none labelA labelB ...
```

This is a possible way to inhibit an earlier use of *PICKUPS* with "labelA, labelB, ..." keyword list. It is sufficient (and necessary) for that, that no keyword in zgoubi.dat data list have "none" as its first *LABEL*.

PLOTDATA **Intermediate output for the PLOTDATA graphic software**

To be documented.

POISSON	**Read magnetic field data from *POISSON* output**		
IC, *IL*	$IC = 1, 2$: print the field map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
BNORM, XN,YN	Field and X-,Y-coordinate normalization coeffs.	3*no dim.	3*E
TITL	Title. Start with "FLIP" to get field map X-flipped		A80
IX, IY	Number of longitudinal and transverse nodes of the uniform mesh	≤ 400, ≤ 200	2*I
FNAME [1]	File name		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	Degree of interpolation polynomial as for *DIPOLE-M*	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* (*e.g.*, ''outpoi.lis'') contains the field map data.
These must be formatted according to the following *FORTRAN* read sequence - details
and possible updates are to be found in the source file `'fmapw.f'` :

```
      I = 0
   11 CONTINUE
      I = I+1
      READ(LUN,101,ERR=99,END=10) K, K, K, R, X(I), R, R, B(I)
101    FORMAT(I1, I3, I4, E15.6, 2F11.5, 2F12.3)
      GOTO II
   10   CONTINUE
```

where $X(I)$ is the longitudinal coordinate, and $B(I)$ is the Z component of the field at a node (I) of the mesh.
K's and R's are variables appearing in the *POISSON* output file outpoi.lis, not used here.

POLARMES	**2-D polar mesh magnetic field map** mid-plane symmetry is assumed		
IC, IL	$IC = 1, 2$: print the map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
BNORM, AN,RN	Field and A-,R-coordinate normalization coeffs.	3*no dim.	3*E
TITL	Title. Start with "FLIP" to get field map X-flipped		A80
IA, JR	Number of angular and radial nodes of the mesh	≤ 400, ≤ 200	2*I
FNAME [1]	File name		A80
ID, A, B, C [, A', B', C', B'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, 2*no dim., cm [,2*no dim., cm, etc.]	I,3*E [,3*E,etc.]
IORDRE	Degree of interpolation polynomial (see *DIPOLE-M*)	2, 25 or 4	I
XPAS	Integration step	cm	E
KPOS	as for *DIPOLE-M*. Normally 2.	1-2	I
If KPOS = 2			
RE, TE, RS, TS		cm, rad, cm, rad	4*E
If KPOS = 1			
DP		no dim.	E

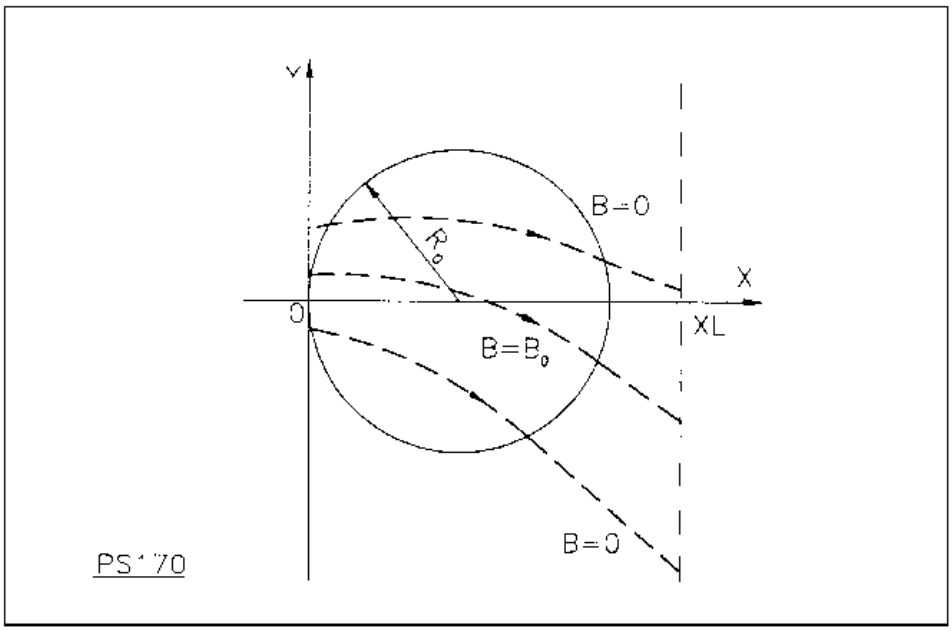
[1] *FNAME* (*e.g.*, spes2.map) contains the field data.
These must be formatted according to the following *FORTRAN* read sequence - details and possible updates are to be found in the source file `'fmapw.f'` :

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD' [,FORM='UNFORMATTED'])
IF (BINARY) THEN
 READ(NL) (Y(J), J=1, JY)
ELSE
       READ(NL,100) (Y(J), J=1, JY)
      ENDIF
100    FORMAT(10 F8.2)
      DO 1 I = 1,IX
 IF (BINARY) THEN
  READ (NL) X(I), (BMES(I,J), J=1, JY)
 ELSE
  READ(NL,101) X(I), (BMES(I,J), J=1, JY)
101  FORMAT(10 F8.1)
       ENDIF
 1   CONTINUE
```

where $X(I)$ and $Y(J)$ are the longitudinal and transverse coordinates and *BMES* is the Z field component at a node (I, J) of the mesh. For binary files, *FNAME* must begin with 'B_' or 'b_'. 'Binary' will then automatically be set to '.TRUE.'

PS170 **Simulation of a round shape dipole magnet**

IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length of the element, radius of the circular dipole, field	2*cm, kG	3*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



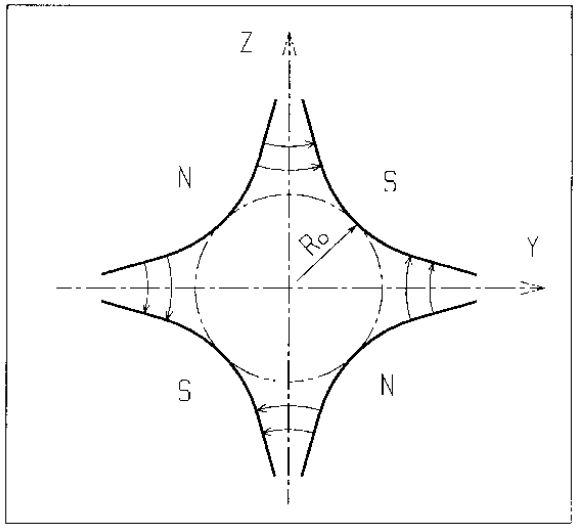
Scheme of the PS170 magnet simulation.

QUADISEX **Sharp edge magnetic multipoles**

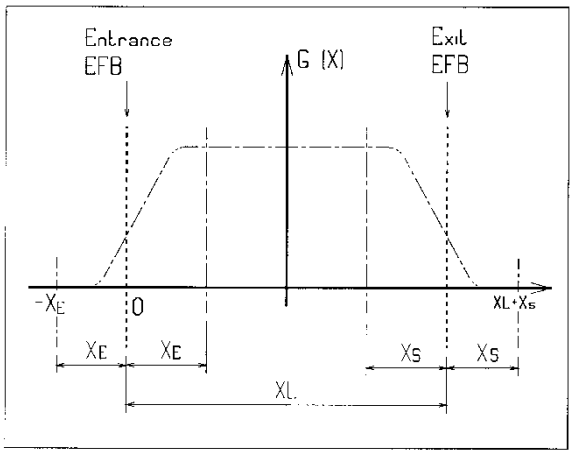
$$B_Z\mid_{Z=0} = B_0\left(1+\frac{N}{R_0}Y+\frac{B}{R_0^2}Y^2+\frac{G}{R_0^3}Y^3\right)$$

IL	$IL=1,2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length of the element ; normalization distance ; field	2*cm, kG	3*E
N, $EB1$, $EB2$, $EG1$, $EG2$	Coefficients for the calculation of B. if $Y>0$: $B=EB1$ and $G=EG1$; if $Y<0$: $B=EB2$ and $G=EG2$.	5*no dim.	5*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

QUADRUPO	**Quadrupole magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0,B_0	Length ; radius and field at pole tip	2*cm, kG	3*E
	Entrance face :		
X_E, λ_E	Integration zone extent ; fringe field extent ($\simeq 2R_0$, $\lambda_E = 0$ for sharp edge)	2*cm	2*E
NCE, $C_0 - C_5$	*NCE* = unused $C_0 - C_5$= Fringe field coefficients such that $G(s) = G_0/(1 + \exp P(s))$, with $G_0 = B_0/R_0$ and $P(s) = \sum_{i=0}^{5} C_i(s/\lambda)^i$	any, 6*no dim.	I, 6*E
	Exit face		
X_S, λ_S	See entrance face	2*cm	2*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



Quadrupole magnet



Scheme of the elements *QUADRUPO, SEXTUPOL, OCTUPOLE, DECAPOLE, DODECAPO* and *MULTIPOL*
(OX) is the longitudinal axis of the reference frame $(0, X, Y, Z)$ of **zgoubi**.
The length of the element is *XL*, but trajectories are calculated from $-X_E$ to $XL + X_S$, by means of automatic prior and further X_E and X_S translations.

REBELOTE **'Do it again'**

NPASS, *KWRIT*, *K[.n]*,	NPASS : Number of runs ; *KWRIT* = 1.1 (resp. 0.0) switches	arbitrary ; 0-1 ;	3*I
[IOPT],	(inhibits) *FORTRAN WRITE*s to .res and to screen ;	0, 99 ; [1[.any]]	2A10
[, Label1 [, Label2]]	K option :	0-1 ; 0, 22, 99	2A10
	$K = 0$: initial conditions (coordinates and spins) are generated following the regular functioning of *[MC]OBJET*. If random generators are used (*e.g.* in *MCOBJET*) their seeds will not be reset.		
	$K = 99$: coordinates at end of previous pass are used as initial coordinates for the next pass ; *idem* for spin components.		
	$K = 99.1$: Label1 is expected, subsequent passes will start from element with Label1 down to *REBELOTE* and so forth ;		
	$K = 99.2$: Label1 and Label2 are expected ; last pass (# NPASS+1) will end at element with Label2 whereupon execution will jump to the keyword next to *REBELOTE* and will be carried out down to *'END'*.		
	IOPT is optional :		
	IOPT = 1 : will change the value of *NPRM* different parameters in one or more of optical elements, using the data list in the next line.		

If IOPT = 1 [1]

NPRM	Number of parameters to be changed at each of the NPASS passes.		I

NPRM **lines with the following sequence (tells parameters concerned, and for each, its successive values) :**

LMNT, *KPRM*, values	*LMNT* : two possibilities, (i) keyword number in zgoubi.dat sequence ; (ii) *KEYWORD[, LABEL]*, i.e., keyword concerned followed optionnally by is label ;	-, -, NV*dim [2]	2*I, NV*E
	KPRM : parameter number under that keyword (similar nomenclature to FIT[2], see page 232) ;		
	'values' are the new parameter values. It can take two forms :		
	- either an explicit list of *NPASS* values		
	- or, first_value:last_value (interval will be (first_value - first_value)(npass-1))		

[1] *IOPT=1* is compatible with use of the *FIT[2]* procedure : *e.g.*, allows successive *FIT*s in a single **zgoubi** run, with successive sets of optical parameter values.

[2] *V* is in current **zgoubi** units in the case of particle coordinates (cm, mrad). It is in MKSA units (m, rad) in the case of matrix coefficients.

• Using **REBELOTE** : An example.

In this example of an energy-recovery electron recirculator based on FFAG arcs, the arguments in the keyword *AUTOREF* and *CAVITE* are changed *20* times by *REBELOTE*, over the 21-pass recirculation process [70]. The role of *AUTOREF* is to mimic a spreader-recombiner, *i.e.*, re-centering the beam on the design FFAG orbit at the start of each one of the 11 accelerated and 10 decelerated ring turns. The RF voltage in *CAVITE* is set positive (accelerating, from 7.944 GeV to 21.16 GeV) during the first 11 passes through the optical structure, and negative (decerating, from 21.16 to 7.944 GeV) over the remaining 10 passes. The voltage value at each pass is adjusted (for an energy of 1.322 GeV per turn on average) so to compensate the energy lost by synchrotron radiation in the arcs.

```
eRHIC ENERGY RECOVERY LINAC RECIRCULATOR WITH FFAG ARCS.
 'MCOBJET'                                                1
57.36635309d3         reference rigidity (kG.cm)
3
2000
2 2 2 2 1 1
-5.360667E-03   5.059706E-3  0. 0. 0.   4.619439E-01  'o'
0. 1   0.   3
0. 1   0.   3
0. 1.  0.     3
123456 234567 345678
 'PARTICUL'                                              2
0.51099892 1.60217653e-19 1.15965218076e-3 0.0 0.0
 'SPNTRK'                                                3
 'FAISCEAU'                                              4
 'SRLOSS'                                                5
1   srLoss
MULTIPOL
1  123456
 'SCALING'                                               6
1  1
MULTIPOL
-1
57.36635309 57.36635309
1           11
 'MARKER'   ARC#S_1                                      7
 'OPTIONS'                                               8
1  1
WRITE OFF
 'MARKER'   MARK      CELLSTART                          9
 'DRIFT'    DRIF      HD                                10
14.547181
 'MULTIPOL' RBEN      BD2                               11
0  .Dip
90. 10. 0. -0.87159105   0. 0. 0. 0. 0. 0. 0. 0.
0. 0. 10.00  4.0  0.800 0.00 0.00 0.00 0.00 0. 0. 0. 0.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0. 0. 10.00  4.0  0.800 0.00 0.00 0.00 0.00 0. 0. 0. 0.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
#30|90|30    Dip  BD2
3   0.0E+00   4.0704703E-01  -1.5071892929E-03
 'DRIFT'    DRIF      D                                 12
29.094362
 'MULTIPOL' RBEN      QF2                               13
0  .Dip
110.  10.  0. 0.86286642  0. 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0. 0. 10.00  4.0  0.800 0.00 0.00 0.00 0.00 0. 0. 0. 0.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0. 0. 10.00  4.0  0.800 0.00 0.00 0.00 0.00 0. 0. 0. 0.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
#30|110|30    Dip  QF2
3   0.  -3.6008008661E-01  -1.8710983347E-03
 'DRIFT'    DRIF      HD                                14
14.547181
 'MARKER'   MARK      CELLEND                           15
-----------------------------------------------------
6*138-1 additional such FD cells, simulating a 6 arc
energy recovery ring, 138 cells per arc.
-----------------------------------------------------
 'MARKER'   ARC#E_6                                   5820
 'OPTIONS'                                           5821
1  1
WRITE ON
 'FAISTORE'                                          5822
zgoubi.fai
1
 'FAISCEAU'                                          5823
 'CAVITE'                                            5824
3
0  0
1.322e9  1.57079632679
 'AUTOREF'                                           5825
4.1
0. -5.168354E-01 4.169759E+00  5.38813280E-01
 'FAISCEAU'                                          5826
 'REBELOTE'                                          5827
20  0.1 99 1
4
AUTOREF 11 -4.7493E-01 -4.1290E-01 -3.3292E-01 -2.3684E-01
-1.26268E-01 -2.59705E-03   ... (a list of 20 values)
AUTOREF 12  3.34619E+00  2.58215E+00  1.87157E+00  1.20916
 5.90197E-01  1.05897E-02   ... (a list of 20 values)
AUTOREF 13 .615682680   .692552080   .769421480  .84629088
.923160280    1.00002970    ... (a list of 20 values)
CAVITE  20  1.337399e9  1.33934e9   1.34012e9   1.34018e9
... (15 more voltage values) ...  -1.30660e9  -1.309577e9
 'SRPRNT'                                            5828
 'END'                                               5829
```

• **Combining REBELOTE and FIT** : An example is given page 234.

RESET **Reset counters and flags**

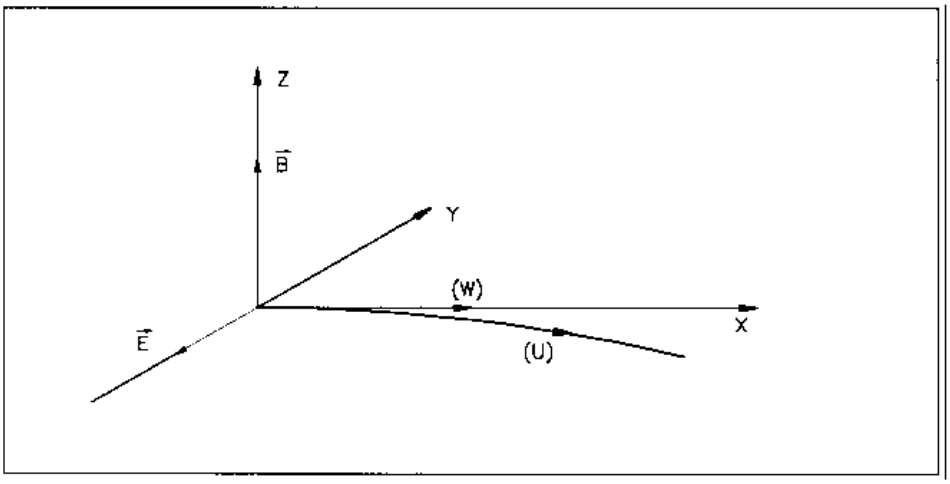
Resets counters involved in *CHAMBR*, *COLLIMA*, *HISTO* and *INTEG* procedures.

Switches off *CHAMBR*, *MCDESINT*, *SCALING* and *SPNTRK* options.

SCALING	**Power supplies and R.F. function generator**		
IOPT, NFAM [, PRINT]	*IOPT* = 0 (inactive) or 1 (active) ; *NFAM* = number of families to be scaled. Occurence of *PRINT* will cause printing of scaling infos in zgoubi.SCALING.out.	0-1, 1-45 [,-]	I1, I2 [,A]
For NF=1, NFAM :	repeat *NFAM* times the following sequence :		
NAMEF [, Lbl [, Lbl]]	Name of family (*i.e.*, keyword of concern) ; up to 2 labels, wild card accepted, in the form *'*myLabel'* or *'myLabel*'*.		A10 [,A10[,A10]]
NT	$NT > 0$: number of timings ; $NT = -1$: field scaling factor updated by *CAVITE* ; $NT = -2$: RF law in *CAVITE* is read from external data file.	-2, -1 or 1-10	I
$SCL(I)$, $I = 1$, NT	Scaling values, and in particular : a single value if $NT = -1$; unused if $NT = -2$: if $NT = 1$ then $SCL(I) = SCL(1)$ whatever the pass number.	relative	NT*E
$TIM(I)$, $I = 1$, NT	Corresponding timings, in units of turns (1 if $NT = -1$; unused if $NT = -2$).	turn number	NT*I

SEPARA [1] **Wien Filter - analytical simulation**

IA, XL, E, B,	$IA = 0$: element inactive $IA = 1$: horizontal separation $IA = 2$: vertical separation ; Length of the separator ; electric field ; magnetic field.	0-2, m, V/m, T	I, 3*E



Horizontal separation between a wanted particle, (W), and an unwanted particle, (U).
(W) undergoes a linear motion while (U) undergoes a cycloidal motion.

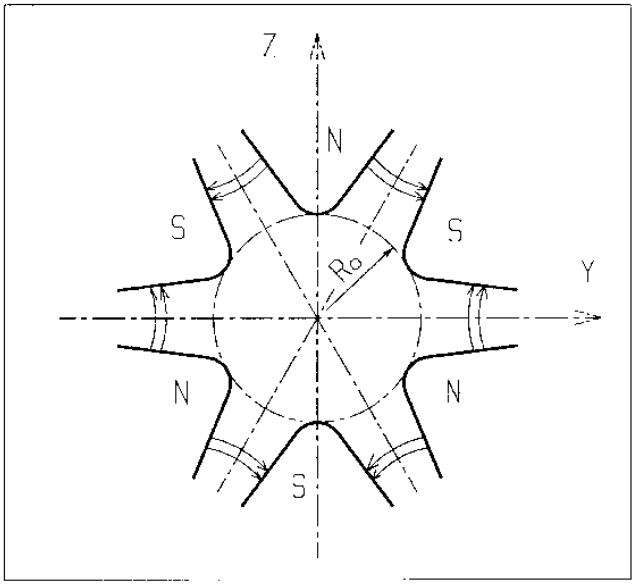
[1] *SEPARA* must be preceded by *PARTICUL* for the definition of mass and charge of the particles.

SEXQUAD **Sharp edge magnetic multipole**

$$B_Z\ |_{Z=0} = B_0 \left(\frac{N}{R_0} Y + \frac{B}{R_0^2} Y^2 + \frac{G}{R_0^3} Y^3 \right)$$

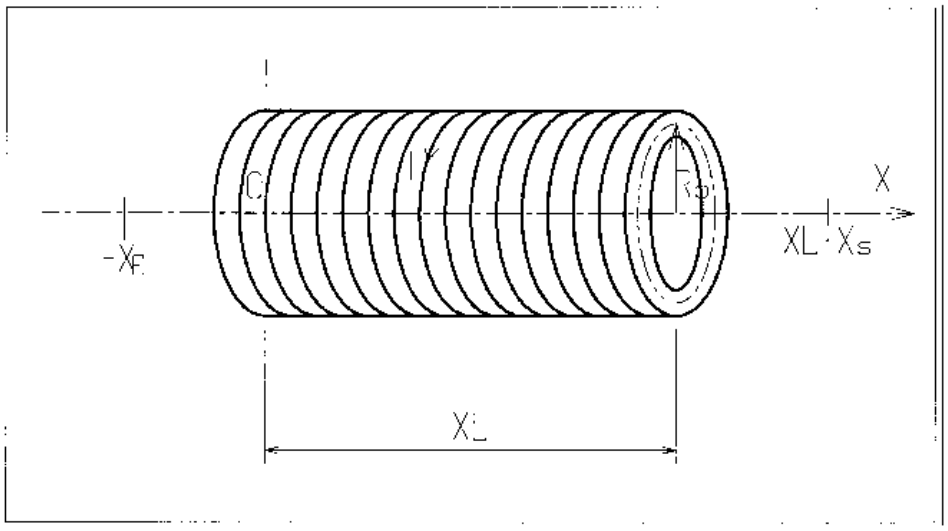
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length of the element ; normalization distance ; field	2*cm, kG	3*E
N, $EB1$, $EB2$, $EG1$, $EG2$	Coefficients for the calculation of B. if $Y > 0$: $B = EB1$ and $G = EG1$; if $Y < 0$: $B = EB2$ and $G = EG2$.	5*no dim.	5*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

SEXTUPOL	**Sextupole Magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0	Length ; radius and field at pole tip of the element	2*cm, kG	3*E
	Entrance face :		
X_E, λ_E	Integration zone ; fringe field extent ($\lambda_E = 0$ for sharp edge)	2*cm	2*E
NCE, $C_0 - C_5$	*NCE* = unused $C_0 - C_5$ = Fringe field coefficients such that $G(s) = G_0/(1 + \exp P(s))$, with $G_0 = B_0/R_0^2$ and $P(s) = \sum_{i=0}^{5} C_i(s/\lambda)^i$	any, 6* no dim.	I, 6*E
	Exit face :		
X_S, λ_S	Parameters for the exit fringe field ; see entrance	2*cm	2*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



Sextupole magnet

SOLENOID	**Solenoid**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, R_0, B_0 *[, MODL]*	Length ; radius ; asymptotic field (=$\mu_0 NI/XL$) ; *MODL=1* : default, r-extrapolation from on-axis field model, *MODL=2* : elliptic integrals method	2*cm, kG [, no dim.]	3*E
X_E, X_S	Entrance and exit integration zones	2*cm	2*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



SPACECHARG	**Space charge**		
LMNT, model [, PRINT]	*LMNT* = ’all’, ’none’, or keyword ; model : ’KV’, ’Gaussian’ ; [optional : print out to zgoubi.SPACECHARG.out].	-, - [,-]	2*A [,A]
λ	Linear charge density	C/m	E

SPINR	**Spin rotation**		
IOPT	Option	0-2	I
If IOPT=0	Element inactive		
X, X	Unused	2*unused	2*E
If IOPT=1	**Axis and spin precession angle values.**		
ϕ, μ	Angle (in (X,Y) plane) between the X-axis and the spin precession axis ; spin precession angle around that axis.	deg, deg	2*E
If IOPT=2	**Given the spin precession axis direction, ϕ, in the horizontal plane, the spin precession angle follows a function of the Lorentz factor :** $\mu(\gamma) = \left(\frac{B}{B_0}\right)^2 \times \left(C_0 + \frac{C_1}{\gamma} + \frac{C_2}{\gamma^2} + \frac{C_3}{\gamma^3}\right)$		
ϕ, B, B_0, C_0, C_1, C_2, C_3	Angle (in (X,Y) plane) between the X-axis and the spin precession axis ; six coefficients that define the spin precession around that axis.	deg.; 6*no dim	7*E

SPNPRNL **Store spin coordinates in file *FNAME***

FNAME [1]	Name of storage file (*e.g.*, zgoubi.spn)	A80

SPNSTORE **Store spin coordinates every IP other pass**

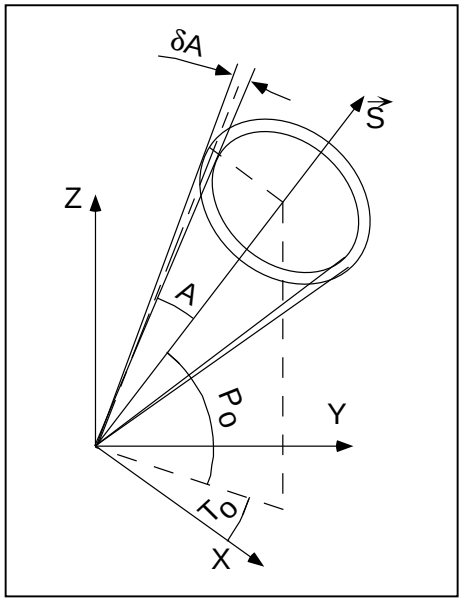
FNAME [1] [,*LABEL(s)*] [2]	Name of storage file (*e.g.*, zgoubi.spn) [; label(s) of the element(s) at the exit of which the store occurs (10 labels maximum)].	A80 [, 10*A10]
IP	Store every IP other pass (when using *REBELOTE* with *NPASS* $\geq IP - 1$).	I

SPNPRT [,PRINT] [,MATRIX] **Print spin coordinates, compute spin matrix**

Print spin coordinates in zgoubi.res, at the location where this keyword is introduced in the structure.
A label '*PRINT*' is optional, if it appears, then local data are stored in the file zgoubi.SPNPRT.Out. That file is opened at the first occurence of *SPNPRT* and left open until **zgoubi** execution is completed.
A label '*MATRIX*' is optional, if it appears, then computation of spin rotation matrix is performed. This however assumes appropriate trajectory and spin sampling (see Sec. 5.5.10, p. 175).

[1] *FNAME* = 'none' will inhibit printing.
[2] If first *LABEL* = 'none' then printing will be inhibited.

SPNTRK [1]	**Spin tracking**		
KSO [.KSO2]	*KSO=0* : spin tracking [switched] off [2] ; *KSO= -1* : spin tracking resumes. Otherwise : as stated below.	-1 or 0 or 1-3 0r 4[.1] or 5	I
If KSO = 1 – 3	*KSO* = 1 (respectively 2, 3) : all particles have their spin automatically set to (1,0,0), longitudinal (respectively (0,1,0), horizontal and (0,0,1), vertical)		
If KSO = 4	Repeat *IMAX* times (corresponding to the *IMAX* particles in '*OBJET*') the following sequence :		
S_X, S_Y, S_Z	X, Y and Z initial components of the initial spin.	3*no dim.	3*E
If KSO = 4.1			
S_X, S_Y, S_Z	X, Y and Z components of the initial spins. These will be assigned to all particles.	3*no dim.	3*E
If KSO = 4.2			
FNAME [3]	File name *(under development)*		A80
If KSO = 5	Random distribution in a cone (see figure) Enter the following two sequences :		
TO, PO	Angles of average polarization direction.	2*rad	2*E
A, δA	A = angle of the cone around (TO, PO) ; δA = standard deviation of distribution around A	2*rad	2*E
IR	Random sequence seed	$\lesssim 10^6$	I



Option *KSO* = 5 : spins are distributed at random within an annular strip with standard deviation δA, centered at an angle A with respect to the axis of mean polarization ($\vec{S}$) defined by T_0 and P_0.

[1] *SPNTRK* must be preceded by *PARTICUL* for the definition of *G* and mass.

[2] Spin tracking can be switched off at any location in zgoubi.dat data list using *KSO=0*, and further away resumed using *KSO= -1*.

[3] *FNAME* contains the spin components. Formatting should be has follows :

TXT, SXI, SYI, SZI, SMI,SX, SY, SZ, SM

with TXT a (arbitrary) 3-character string, SXI, SYI, SZI, SMI the initial spin components and their modulus (unused), SX, SY, SZ, SM the current spin components (these will be tracked further) and their modulus(unused).

SRLOSS	**Synchrotron radiation energy loss**		
KSR[.i] [,KSOK]	On/off switch ; $i = 1$ for info output into `zgoubi.SRLOSS.Out` ; Sokolov-Ternov effect switch, optional (1/0 for on/off, default is off).	$0\ or\ 1[.1]$	I[.I], I
STR1[{lblst}] *[, STR2 [,List]]*	Apply SR loss to *STR1 = 'ALL'* or *'all'* or a particular KEYWORD ; {*lblst*}={*lbl1, lbl2, ...,lbl5*} : a list of up to 5 labels to which this applies. Optional : *STR2 = 'scale'* will scale fields following energy loss ; *List* : a list of up to 10 keywords to which this scaling applies.		A[1-5A] [,A] [,1-10A]
Option, seed	Option = 0 or 1 : SR causes dp only ; option = 2 : SR causes dp and angle kick (under development).	$1-3, > 10^5$	2*I

SRPRNT **Print SR loss statistics** into zgoubi.res

SYNRAD	**Synchrotron radiation spectral-angular densities**		
KSR	Switch 0 : inhibit SR calculations 1 : start 2 : stop	0-2	I
If KSR = 0			
$D1$, $D2$, $D3$	Dummies		3*E
If KSR = 1			
$X0$, $Y0$, $Z0$	Observer position in frame of magnet next to *SYNRAD*	3*m	3*E
If KSR = 2			
ν_1, ν_2, N	Frequency range and sampling	2*eV, no dim.	2*E, I

SYSTEM	**System call**		
NCMD	The number of calls to follow.	≥ 0	I

NCMD lines follow, one command per line.

• **SYSTEM** : An example (an additional example can be found page 234)

The first occurence of the command (top region in the data list below) allows establishing links from remote folders to the current one (the one in which **zgoubi** is presently run). These folders happen to contain files appearing in the *SCALING* command, as well as OPERA field maps of the siberian snakes used in subsequent *TOSCA* commands.

The second occurence of the command (bottom region in the data list below) allows saving zgoubi.res output file as resulting from the **zgoubi** run, under a different name.

```
AGS, polarized protons. 2 snakes. t = 145 ms.
'GETFITVAL'
fitVals.data
'OBJET'     LBL_OBJfit
7069.3668040036146
5
.01 .01 .01 .01 0. .0001
  0.  0.  0.  0.  0.  1.    'o'
'FAISCEAU'
'SYSTEM'
 2
ln -sf /rap/lattice_tools/zgoubi/AgsZgoubiModel/snakeFieldMaps/TOSC3D/Csnk3D .
ln -sf /rap/lattice_tools/zgoubi/AgsZgoubiModel/snakeFieldMaps/TOSC3D/Wsnk3D .
'SCALING'   LBL_SCLfit
1   21
AGSMM *AF *BF *CF  !# of params 2Bchanged. dB0 (FIT#3) dB1 (FIT#4)      dB2
-1                  3                      13  3E-20   14 -1.473595E-03 15  8.01942
1.00000000     ! (FIT #6)
1
AGSMM *AD *BD *CD  !# of params 2Bchanged. dB0 (FIT#9) dB1 (FIT#10)     dB2
-1                  3                      13  9E-20   14 -1.011857E-03 15 -1.70597
1.000000      ! (FIT#12)
1
AGSQUAD  QH_*   !# of params 2Bchanged.      (FIT#15)
-1               1                      15   0.0
1.000000
1
AGSQUAD  QV_*  QP_*      !# of params 2Bchanged.       (FIT#15)
-1                       1                      15    0.0
1.000000
1
MULTIPOL   QJUMP_*
-1
7.06936680E+00
1
MULTIPOL   SXH_*
-1
7.06936680E+00
1
MULTIPOL   SXV_*
-1
7.06936680E+00
1
AGSMM MM_F08CD MM_F09BF MM_G02BF MM_G03CD MM_G16AD MM_G17CF  ! blw1 / G9  bump
-1          1        22    -.0762e-99                ! Amp.  (FIT #31)
1.0
1
AGSMM MM_H04CD MM_H05AF MM_H18CF MM_H19BD MM_I12BD MM_I13CF  ! blw1 / H11 bump
-1          1        22    -.06093e-99               ! Amp.  (FIT #35)
1.0
1
MULTIPOL COH1
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal
1 4
MULTIPOL COV1
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal
1 5
MULTIPOL COH2
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal
1 6
MULTIPOL COV2
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal
1 7
MULTIPOL WOH1
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal_51.8_0.0
1 4
MULTIPOL WOV1
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal_51.8_0.0
1 5
MULTIPOL WOH2
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal_51.8_0.0
1 6
MULTIPOL WOV2
1.10
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal_51.8_0.0
1 7
CHANGREF WSNKE
1.12
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal_51.8_0.0
1 1  1 3
CHANGREF WSNKO
1.12
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal_51.8_0.0
1 1  1 2
CHANGREF CSNKE
1.12
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal
1 1  1 3
CHANGREF CSNKO
1.12
./Csnk3D/Hlx68.2_Sol42.3/CHREF_+_dipolCORR.scal
1 1  1 2
```

```
'MARKER'   #Start
'OPTIONS'
1 1  ! options
WRITE OFF
'AGSMM'  MM_A01BF
0
3 0 0   0.00000E+00  1.00000E+00  1.00000E+00  1.0
2.1  1 0.  1 0.
0. 0. 10.00  4.0  0.800 0.00 0.00 0.00 0.00 0. 0. 0. 0.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0. 0. 10.00  4.0  0.800 0.00 0.00 0.00 0.00 0. 0. 0. 0.
4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
3.0  Dip MM_A01BF
4 0. 0. 0. 0. 0.
[......]
'TOSCA'
0  0
1.e-3 100. 100. 100.
HEADER_4 csnake
281 29 29  15.2 .682 .423
./Csnk3D/Hlx68.2_Sol42.3/b_table_for_Helix_3T.tab
./Csnk3D/Hlx68.2_Sol42.3/b_table_for_Solen.tab
0 0 0 0
2
.1
2  0.  .00  0.  0.
[......]
'TOSCA'
0  0
1.e-3 100. 100. 100.
HEADER_4 csnake
281 29 29  15.2 .518 1.e-30
./Csnk3D/Hlx68.2_Sol42.3/b_table_for_Helix_3T.tab_2
./Csnk3D/Hlx68.2_Sol42.3/b_table_for_Solen.tab_2
0 0 0 0
2
.1
2  0.  .00  0.  0.
[......]
'MARKER'   #End
'FAISCEAU'
'OPTIONS'
1 1  ! options
WRITE ON
'FAISCEAU'
'MATRIX'
1   11
'SYSTEM'
1
cp zgoubi.res zgoubi.res_save
'END'
```

TOSCA	**2-D and 3-D Cartesian or cylindrical mesh field map** (page 153 in PART A)		
IC, *IL*	$IC = 1, 2$: print the map $IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2; 0-2$[\times 10^n]$, 7	2*I
BNORM, XN,YN, ZN	Field and X-,Y-,Z-coordinate normalization coefficients. Convert values as read from map file, to kG and cm or rad.	4*UnitConv.	4*E
TITL	Title. Include "FLIP" to get field map X-flipped. Include "HEADER n" if *FNAME* starts with $n \geq 1$ header lines. Include "ZroBXY" to force $B_X = B_Y = 0$ at all Z=0 (only applies for *MOD*=15 and *MOD*=24). Include "RHIC_helix" to normalize *BNORM* to measured value (B field versus helix current, in program toscac.f).		A80
IX, IY, IZ, [1] *MOD[.MOD2 [, a(i),i= 1,MOD2]]*	Number of nodes of the mesh in the X, Y and Z directions, $IZ = 1$ for a 2-D map ; *MOD* and *MOD2* : field map style, see table next page.	$\leq$*MXX*, $\leq$*MXY*, $\leq$*IZ*, [0-22.1-9], -	3*I, I[.I [, MOD2*E]
Next NF lines : *FNAME* [3]	Map file name(s), one line per name. If *MOD*=0 : $NF = 1 + [IZ/2]$, the NF 2-D maps are for $0 \leq Z \leq Z_{max}$, they are symmetrized with respect to the $Z(1) = 0$ plane. If *MOD*=1 : $NF = IZ$, no symmetry assumed ; $Z(1) = Z_{max}$, $Z(1 + [IZ/2]) = 0$ and $Z(NF) = -Z_{max}$. If *MOD*=12 : a single *FNAME* file contains the all 3-D volume. *MOD=15, 20-22*, etc. : see table next page.		A80
ID, A, B, C [, A', B', C', A'', etc., if $ID \geq 2$]	Integration boundary. Ineffective when $ID = 0$. ID = -1, 1 or ≥ 2 : as for *CARTEMES*	≥ -1, cm, 2*no dim. [,*idem*]	I,3*E [,3*E,etc.]
IORDRE	If $IZ = 1$: 2, 25 or 4 as in *CARTEMES* ; unused if $IZ \neq 1$.	2, 25 or 4	I
XPAS	Integration step	cm	E
If Cartesian mesh (see MOD) :			
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt	1-2, 2*cm, rad	I, 3*E
If polar mesh :			
KPOS	as for *POLARMES*. Normally 2.	1-2	I
If KPOS = 2 :			
RE, TE, RS, TS		cm, rad, cm, rad	4*E

[1] *MXX, MXY, IZ* may be changed, they are stated in the include file `PARIZ.H`.

[2] Case of 2-D field maps : Each file *FNAME(K)* contains the field specific to elevation $Z(K)$ and must be formatted according to the following *FORTRAN* read sequence (that usually fits *TOSCA* code *OUTPUTS* - details and possible updates are to be found in the source file `'fmapw.f'`) :

```
DO JF = 1, NF
  OPEN (UNIT = NL, FILE = FNAME(JF), STATUS = 'OLD' [,FORM='UNFORMATTED'])
  DO J = 1,  JY ;      DO I = 1, IX
     READ(NL,*) Y(J), Z(JF), X(I), BY(J,I), BZ(J,I), BX(J,I)        node coordinates, field components at node
  ENDDO ;    ENDDO
  NL = NL + 1

ENDDO
```

BX and BY are assumed zero at all nodes of the 2-D mesh, regardless of BX(J,1,I), BY(J,1,I) values.
Case of 3-D field maps :

```
DO JF = 1, NF
  OPEN (UNIT = NL, FILE = FNAME(JF), STATUS = 'OLD' [,FORM='UNFORMATTED'])
  DO J = 1,  JY ; DO K = 1, KX ; DO I = 1, IX
     READ(NL,*) Y(J), Z(K), X(I), BY(J,K,I), BZ(J,K,I), BX(J,K,I)        node coordinates, field components at node
  ENDDO ; ENDDO ; ENDDO
  NL = NL + 1

ENDDO
```

[3] For binary files, *FNAME* must begin with 'B_' or 'b_'.

• **The various *IZ, MOD* and *MOD2* possibilities, when using *TOSCA*.**

IZ : number of nodes of the *complete* field map along the Z direction (IZ=1 for 2D)
MOD, MOD2 : determine the coordinate system, symmetries, reading format and column sequence, etc.
NF : number of field map input data files to be declared. Always include mid-plane map.
Expected columns : formatting of the coordinates and field data columns in the field map data file(s)
'Exemple' example folder : examples of **zgoubi** runs using field maps can be found in the subfolders of zgoubi-code/exemples/KEYWORDS/TOSCA/cartesian (case MOD≤19) or zgoubi-code/exemples/KEYWORDS/TOSCA/cylindrical (case MOD≥20). The rightmost column below indicates the subfolder of concern, following *(IZ, MOD, MOD2)* options.

MOD ≤ 19 : Cartesian mesh

IZ	MOD	.MOD2		NF	Expected columns	Example folder
1	0, 1	none or .1, .2, .3	**2-D** map. File contains $B_Z(X,Y)\|_{Z=0}$, mid-plane antisymmetry assumed. Several different reading formats (see fmapw.f/fmapr3).	1	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_1-0-none/ (GSI KAOS spectrometer)
1	3	none *or* .1	**2-D** map. Used for AGS main magnet. If MOD2=1, $B_Z(X,Y,Z=0)$ field is perturbed by $(1+n_1\,Y+n_2\,Y^2+n_3\,Y^3)$ factor.	1	Special - see example	AGS/usingMainMagnetsMaps (AGS with main magnet maps)
1	15	.1 − .4	**2-D** map. Up to 4 files to be combined linearly into a new map : field at all node of new map is $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Mid-plane antisymmetry is assumed : each file has to contain $B_Z(X,Y,Z=0)$.	1 − 4	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_1-15-.1-.4 (EMMA FFAG cell)
>1	0	none	**3-D** map. Files span upper half of magnet, one per (X,Y)$\|_{0\le Z\le Z_{max}}$ plane including median plane, mid-plane antisymmetry assumed.	1+IZ/2	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_gt1-0-none/ (GSI KAOS spectrometer)
2p+1 $p\ge 1$	1	none	**3-D** map. Files span full magnet volume, one file per (X,Y) plane, no symmetry assumed.	IZ	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_gt1-1-none/ (AGS warm helix snake)
>1	12	none	**3-D** map. Single file, upper half of magnet, mid-plane antisymmetry assumed.	1		
>1	12	.1	**3-D** map. Single file, whole magnet volume, no symmetry assumed.	1	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_gt1-12-.1/ (AGS warm helix snake)
>1	12	.2	**3-D** map. Single file, 1/8th of the magnet, symmetry wrt. $(X,Y)_{Z=0}$, $(X,Z)_{Y=0}$, $(Y,Z)_{X=0}$ planes.	1		
2p+1 $p\ge 1$	15	.1 − .4	**3-D** map. Up to 4 files to be combined linearly into a new map, field at all node of new map is $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file has to contain $B_{X,Y,Z}(X,Y,Z)$ data over *IZ* equally Z-spaced (X,Y) planes (no symmetry assumed).	1 − 4		
2p+1 $p\ge 1$	16	.1 − .4	**3-D** map. Fields from up to 4 maps to be combined linearly into a new field value at particle location, $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file has to contain $B_{X,Y,Z}(X,Y,Z)$ data over *IZ* equally Z-spaced (X,Y) planes.	1 − 4	UNDER DEVELOPMENT	

MOD ≥ 20 : Cylindrical mesh

IZ	MOD	.MOD2		NF	Expected columns	Example folder
1	25	.1 − .4	**2-D** map. Up to 4 files to be combined linearly into a new map : at all node of new map $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file contains mid-plane $B_Z(X,Y,Z=0)$ data, mid-plane antisymmetry is assumed.	1 − 4	Y, Z, X, BY, BZ, BX	IZ-MOD-.MOD2_1-15-.1-.4 (EMMA FFAG cell)
>1	20, 21		**3-D** map. Single file. MOD=20 : 1/4 magnet, cylindrical symmetry with respect to (Y,Z) plane and antisymmetry wrt (X,Y) plane. MOD=21 : another type of symmetry (to be documented - see fmapw.f).	1	Y(r,θ), Z, X(r,θ), BY, BZ, BX	IZ-MOD-.MOD2_gt1-20 (KEK 150 MeV FFAG)
2p+1 $p\ge 0$	22, 23	.1 − .4	**2D** or **3-D** map. Mid-plane antisymmetry assumed. Up to 4 files can be combined linearly into a new one, $Z\ge 0$ half-magnet volume each : field at all node of new map is $\vec{B}=\sum_{i=1}^{i=MOD2} a_i\,\vec{B}_i$. Each file has to contain $B_{X,Y,Z}(X,Y,Z)$ data over *IZ* equally Z-spaced (X,Y) planes. MOD=23 : special, test code (see fmapw.f).	1 − 4	Y(r,θ), Z, X(r,θ), BY, BZ, BX	
2p+1 $p\ge 1$	24		**3-D** map, full volume. No symmetry assumed.	1	θ, R, Z, B_θ, BR, BZ	

TRANSMAT	**Matrix transfer**		
IORDRE	Transfer matrix order	1-2	I
XL	Length (ineffective, for updating)	m	E
For $IA = 1, 6$:			
$R(IA, IB)$, $IB = 1,\ 6$	First order matrix	m, rad	6 lines 6*E each
If IORDRE = 2	Following records *only* if *IORDRE* = 2		
$T(IA, IB, IC)$,	Second order matrix, six 6*6 blocks	m, rad	36 lines 6*E each

TRAROT	**Translation-Rotation**		
$TX, TY, TZ,$ RX, RY, RZ	Translations, rotations	3*m, 3*rad	6*E

TWISS **Calculation of periodic optical parameters. Log to zgoubi.TWISS.out**

KTW[.KTW2], FacD, FacA *[, coupled]*	$KTW = 0/1/2/3$: Off / as *MATRIX* / add computation of chromaticities / add computation of anharmonicities. $KTW2$: default is initial computation of closed orbit coordinates, $KTW2 = 1$ will avoid that. $\mathrm{FacD} \times \mathrm{D} = \delta\mathrm{p}/\mathrm{p}$ value applied, with D the momentum sampling in OBJET ; $FacA$: unused. "*coupled*", in the periodic case, will cause use of coupled formalism.	0-3[.1], 2*any	I1[.I1], 2*E

• ***Example***

```
'OBJET'
20015.55                          ! 6 GeV electrons.
5                                 ! Will generate 11 particles.
 .001 .001 .001 .001 0. .0001     ! Coordinate sampling for matrix computation : $delta_Y,
0. 0. 0. 0. 0. 1.                 ! delta_T, delta_Z, delta_P, delta_S (unused), delta_D$.
.............................
zgoubi.dat optics list in between
.............................
 'TWISS'
 2 1. 1.                          ! KTW = 3, FacD = 1
 'END'
```

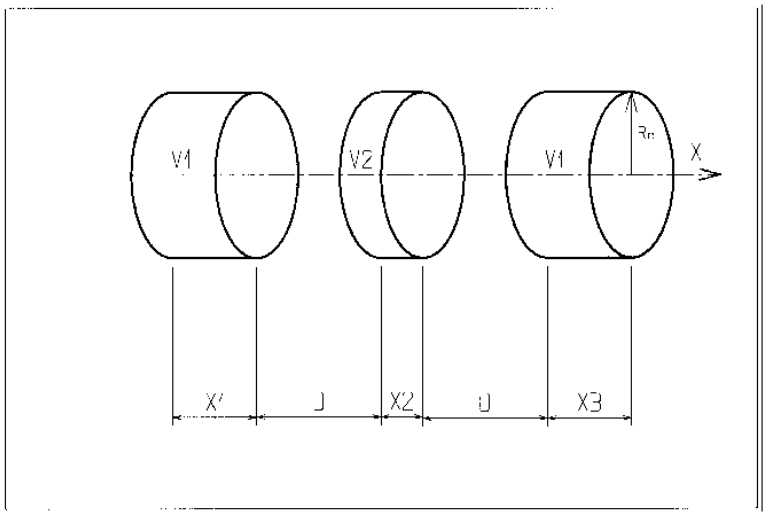
"*KTW=3*" under *TWISS* will cause 3 successive executions of zgoubi.dat and will result in delivery (print out to zgoubi.res) of

- the on-momentum matrix of the optical structure,
- off-momentum matrices at $\frac{dp}{p} = \pm FacD * \delta D$,
- the Twiss parameters in the hypothesis of a stable periodic structure,
- the momentum compaction, chromaticities, etc.

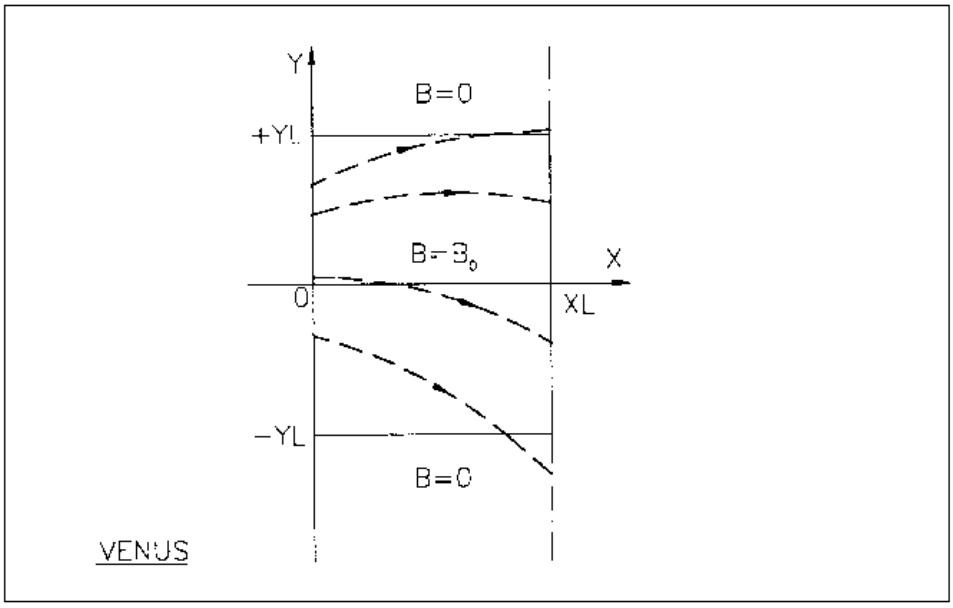
UNDULATOR **Undulator magnet**

Under development, to be documented

UNIPOT	**Unipotential electrostatic lens**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
X_1, D, X_2, X_3, R_0	Length of first tube ; distance between tubes ; length of second and third tubes ; radius	5*m	5*E
V_1, V_2	Potentials	2*V	2*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



VENUS	**Simulation of a rectangular dipole magnet**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, YL, B_0	Length ; width = $\pm YL$; field	2*cm, kG	3*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E



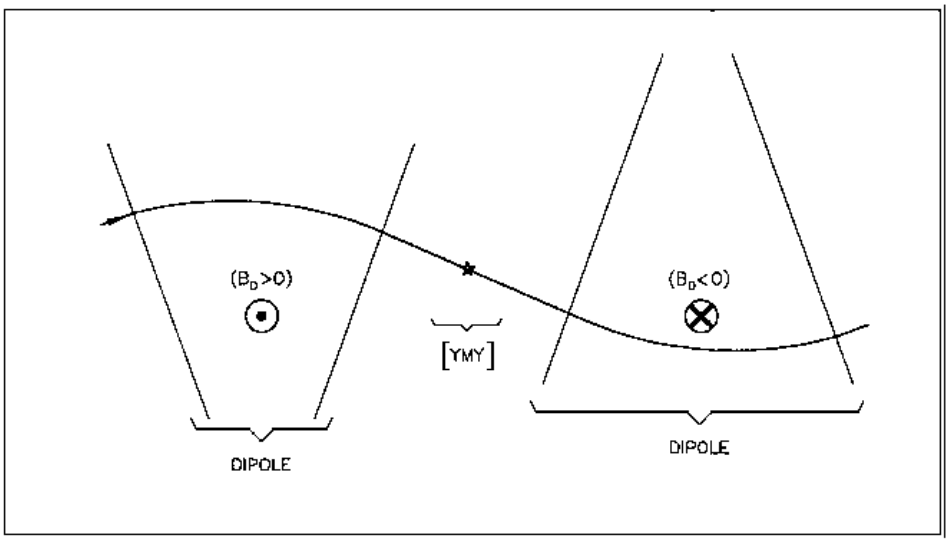
Scheme of *VENUS* rectangular dipole.

WIENFILT [1]	**Wien filter**		
IL	$IL = 1, 2[\times 10^n],\ 7$: print coordinates, fields, etc., step-by-step, in zgoubi.res (1), zgoubi.plt (2), zgoubi.impdev.out (7).	0-2$[\times 10^n]$, 7	I
XL, E, B, HV	Length ; electric field ; magnetic field ; option : element inactive ($HV = 0$) horizontal ($HV = 1$) or vertical ($HV = 2$) separation	m, V/m, T, 0-2	3*E, I
	Entrance face :		
$X_{\rm E}$, λ_{E_E}, λ_{B_E}	Integration zone extent ; fringe field extents, E and B respectively ($\simeq$ gap height)	3*cm	3*E
C_{E0}–C_{E5}	Fringe field coefficients for E	6*no dim.	6*E
C_{B0}–C_{B5}	Fringe field coefficients for B	6*no dim.	6*E
	Exit face :		
X_S, λ_{E_S}, λ_{B_S}	See entrance face	3*cm	3*E
C_{E0}–C_{E5}		6*no dim.	6*E
C_{B0}–C_{B5}		6*no dim.	6*E
XPAS	Integration step	cm	E
KPOS, XCE, YCE, ALE	*KPOS*=1 : element aligned, 2 : misaligned ; shifts, tilt.	1-2, 2*cm, rad	I, 3*E

[1] Use *PARTICUL* to declare mass and charge.

YMY **Reverse signs of Y and Z axes**

Equivalent to a 180° rotation with respect to X-axis



The use of YMY in a sequence of two dipoles of opposite signs.

PART C

Examples of input data files and output result files

INTRODUCTION

Several examples of the use of **zgoubi** are given here. They show the contents of the input and output data files, and are also intended to help understanding some subtleties of the data definition.

Example 1: checks the resolution of the QDD spectrometer SPES 2 of SATURNE Laboratory [62], by means of a *Monte Carlo initial object* and an *analysis of images* at the focal plane with histograms. The *measured field maps* of the spectrometer are used for that purpose. The layout of SPES 2 is given in Fig. 48.

Example 2: calculates the *first and second order transfer matrices* of an 800 MeV/c kaon beam line [63] at each of its four foci: at the end of the first separation stage (vertical focus), at the intermediate momentum slit (horizontal focus), at the end of the second separation stage (vertical focus), and at the end of the line (double focusing). The first bending is represented by its *3-D map* previously calculated with the TOSCA magnet code. The second bending is simulated with *DIPOLE*. The layout of the line is given in Fig. 49.

Example 3: illustrates *the use of MCDESINT and REBELOTE* with a simulation of the *in-flight decay*

$$K \longrightarrow \mu + \nu$$

in the SATURNE Laboratory spectrometer SPES 3 [34]. The angular acceptance of SPES 3 is ± 50 mrd horizontally and ± 50 mrd vertically; its momentum acceptance is ± 40%. The bending magnet is simulated with *DIPOLE*. The layout of SPES 3 is given in Fig. 50.

Example 4: illustrates the operation of *the fitting procedure*: a quadrupole triplet is tuned from -0.7/0.3 T to field values leading to transfer coefficients R12=16.6 and R34=-.88 at the end of the beam line. Other example can be found in [64].

Example 5: shows the use of the *spin and multi-turn tracking procedures*, applied to the case of the SATURNE 3 GeV synchrotron [7, 10, 65]. Protons with initial vertical spin ($\vec{S} \equiv \vec{S}_Z$) are accelerated through the $\gamma G = 7 - \nu_Z$ depolarizing resonance. For easier understanding, some results are summarized in Figs. 52, 53 (obtained with the graphic post-processor **zpop**, see Part D).

Example 6: shows *ray-tracing through a micro-beam line* that involves *magneto-electrostatic quadrupoles* for the suppression of second order (chromatic) aberrations [6]. The extremely small beam spot sizes involved (less than 1 micrometer) reveal the high accuracy of the ray-tracing (Figs. 54).

Many more examples can be found on the **zgoubi** development web site [5].

1 MONTE CARLO IMAGES IN SPES 2

Former Saclay/SATURNE and CERN QDD mass spectrometer [62].

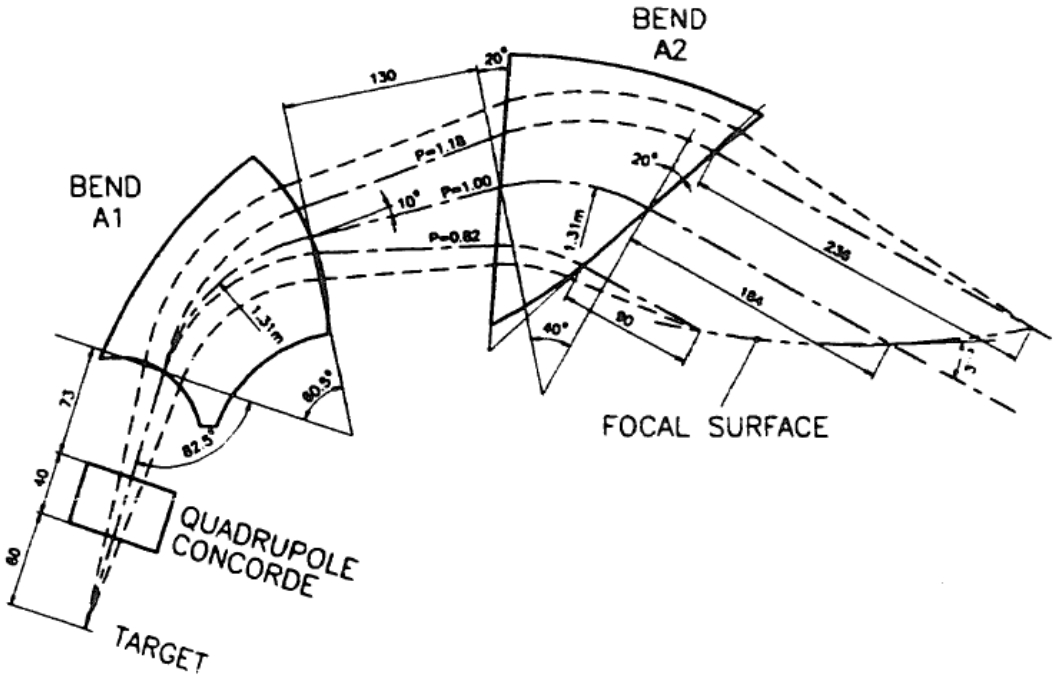


Figure 48: SPES 2 Layout.

zgoubi data file.

```
SPES2 QDD SPECTROMETER, USING FIELD MAPS; MONTE-CARLO OBJECT WITH MOMENTUM GRID.
 'MCOBJET'                                                              1
 2335.                              REFERENCE  RIGIDITY.
 2                                  DISTRIBUTION  IN  GRID.
 10000                              NUMBER  OF  PARTICLES.
 1   1   1   1   1   1              UNIFORM DISTRIBUTIONS
 0.  0.  0.  0.  0.  1.             CENTRAL  VALUES  OF  BARS.
 1   1   1   1   1   5              NUMBER  OF  BARS  IN  MOMENTUM.
 0.  0.  0.  0.  0.  .001           SPACE  BETWEEN  MOMENTUM  BARS.
 0.  50.e-3 0.  50.e-3 0.  0.       WIDTH  OF  BARS.
 1.  1.  1.  1.  1.  1.             SORTING CUT-OFFS (UNUSED)
 9   9. 9. 9. 9.                    FOR P(D) (UNUSED)
 186387 548728 472874               SEEDS.
  'HISTO'                                                               2
  1   .997   1.003  80  1          HISTO  OF  D.
  20  'D'  1  'Q'
  'HISTO'                                                               3
  3   -60.  60.     80  1          HISTO  OF  THETA0.
  20  'T'  1  'Q'
  'HISTO'                                                               4
  5   -60.  60.     80  1          HISTO  OF  PHI0.
  20  'P'  1  'Q'
 'DRIFT'                                                                5
 41.5
 'CARTEMES'                         QUADRUPOLE  MAP.                    6
  0  0                              IC  IL.
  -.96136E-3  1. 1.                 BNORM, XNorm,YNorm
 ++++  CONCORDE ++++
  39  23                            IX  IY.
  concord.map                       field map file name, quadrupole
  0  0 0 0                          NO  LIMIT  PLANE.
 2                                  IORDRE.
 2.5                                XPAS.
 2 0 0 0                            KPOS.
 'DRIFT'                                                                7
 21.8
 'CHANGREF'                         POSITIONING  OF  THE                8
 0. 32.5 -35.6                      1-ST  BENDING.
 'CARTEMES'                                                             9
  0 0
  1.04279E-3  1. 1.
 ++++  A1 ++++
  117  52
  a1.map                            field map file name, first dipole
  0  0 0 0
 2
 2.5
 2 0 0 0
 'CHANGREF'                         POSITIONING  OF  THE                10
 0. -28.65  -27.6137                EXIT  FRAME.
 'DRIFT'                                                                11
 33.15
 'CHANGREF'                         POSITIONING  OF  THE                12
 0. 27.5  -19.88                    2-ND  BENDING.
 'CARTEMES'                                                             13
  0 0
  1.05778E-3  1. 1.
 ++++  A2 ++++
  132  80
  a2.map                            field map file name, second dipole
  0  0 0 0
 2
 2.5
 2 0 0 0
 'CHANGREF'                         POSITIONING  OF  THE                14
 41.  -81.  -21.945                 EXIT  FRAME.
 'DRIFT'                                                                15
 3.55
  'HISTO'                           HISTO  OF  Y :                      16
  2  -.5  2.     80  1              SHOWS  THE  RESOLUTION
  20  'Y'  1  'Q'                   OF  THE  SPECTROMETER.
 'END'                                                                  17
```

Excerpt from zgoubi.res : histograms of initial beam coordinates.

```
******************************************************************************************
     2  HISTO

                    HISTOGRAMME  DE  LA  COORDONNEE    D
                    PARTICULES  PRIMAIRES  ET  SECONDAIRES
                    DANS  LA  FENETRE :   0.9970     /    1.003
                    NORMALISE

  20
  19
  18
  17                  D            D            D            D            D
  16                  D            D            D            D            D
  15                  D            D            D            D            D
  14                  D            D            D            D            D
  13                  D            D            D            D            D
  12                  D            D            D            D            D
  11                  D            D            D            D            D
  10                  0            0            0            0            0
   9                  D            D            D            D            D
   8                  D            D            D            D            D
   7                  D            D            D            D            D
   6                  D            D            D            D            D
   5                  D            D            D            D            D
   4                  D            D            D            D            D
   3                  D            D            D            D            D
   2                  D            D            D            D            D
   1                  D            D            D            D            D
          123456789012345678901234567890123456789012345678901234567890123456789012345678901
                    2         3         4         5         6         7         8         9


       TOTAL  COMPTAGE               :   10000  SUR  10000
       NUMERO   DU  CANAL  MOYEN     :      51
       COMPTAGE  AU   "      "       :    2038
       VAL. PHYS. AU  "      "       :  1.000
       RESOLUTION  PAR  CANAL        :  7.500E-0
       PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                    COMPTAGE =  10000  PARTICULES
                    MIN =  0.9980    , MAX =   1.002    , MAX-MIN =  4.0000E-03
                    MOYENNE =   1.000
                    SIGMA =  1.4108E-03

 TRAJ 1 IEX,D,Y,T,Z,P,S,time :  1  0.9980  0.000 -30.24   0.000   44.63   0.0000   0.0000

******************************************************************************************
     3  HISTO

                    HISTOGRAMME  DE  LA  COORDONNEE  THETA
                    PARTICULES  PRIMAIRES  ET  SECONDAIRES
                    DANS  LA  FENETRE :   -60.00     /    60.00     (MRD)
                           NORMALISE

  20
  19
  18
  17                                                                  T            T
  16                  T                        T           T            T
  15                T T    T      TT         T      T   T   T  T T   T       TT TT
  14               TT T   TT   T  TTTT TT    T     TTTTTT TTTT TTTT  TT T   TTT TTT
  13              TTTTTT TTTTTTT TTTTT TTT TTT T   TTTTTTTTTTTTTTTT TTTTT   TTTTTTTT
  12              TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
  11             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
  10             000000000000000000000000000000000000000000000000000000000000000000000
   9             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   8             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   7             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   6             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   5             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   4             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   3             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   2             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
   1             TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT

          123456789012345678901234567890123456789012345678901234567890123456789012345678901
                    2         3         4         5         6         7         8         9


       TOTAL  COMPTAGE               :   10000  SUR  10000
       NUMERO   DU  CANAL  MOYEN     :      51
       COMPTAGE  AU   "      "       :     128
       VAL. PHYS. AU  "      "       :  3.331E-15 (MRD)
       RESOLUTION  PAR  CANAL        :   1.50     (MRD)

       PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                    COMPTAGE =  10000  PARTICULES
                    MIN =  -49.99    , MAX =   50.00    , MAX-MIN =   99.98     (MRD)
                    MOYENNE =  0.3320     (MRD)
                    SIGMA =   29.04     (MRD)

 TRAJ 1 IEX,D,Y,T,Z,P,S,time :  1  0.9980   0.000  -30.24  0.000  44.63   0.0000   0.0000
******************************************************************************************
```

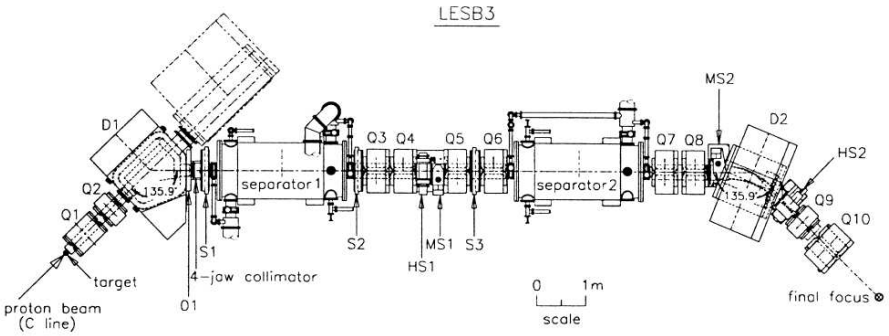
```
*******************************************************************************************
     4  HISTO

                    HISTOGRAMME  DE  LA  COORDONNEE  PHI
                    PARTICULES  PRIMAIRES  ET  SECONDAIRES
                    DANS  LA  FENETRE :   -60.00     /    60.00     (MRD)
                    NORMALISE

   20
   19
   18
   17                                    PP       P   P P
   16               P P P   PP           PPPPP   PPPP PPPP  PPP          PPPP P P
   15              PPPP PPP PPP    PP   P  PPPPPP PPPPPPPPPP PPPP PP  P  PPPPPPP P
   14              PPPP PPP PPPP   PP PPPPPPPPPP PPPPPPPPPPPPPPP PPP PP PPPPPPPPP
   13              PPPP PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   12              PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   11             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   10             000000000000000000000000000000000000000000000000000000000000000000
    9             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    8             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    7             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    6             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    5             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    4             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    3             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    2             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
    1             PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP

          123456789012345678901234567890123456789012345678901234567890123456789012345678901
                   2         3         4         5         6         7         8         9


       TOTAL  COMPTAGE                :   10000  SUR  10000
       NUMERO   DU  CANAL  MOYEN      :      51
       COMPTAGE  AU   "      "        :     163
       VAL. PHYS. AU  "      "        :  3.331E-15 (MRD)
       RESOLUTION  PAR  CANAL         :   1.50

       PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                     COMPTAGE =  10000  PARTICULES
                     MIN =  -50.00    , MAX =   49.99    , MAX-MIN =   99.99     (MRD)
                     MOYENNE =  0.2838     (MRD)
                     SIGMA =   28.75     (MRD)

 TRAJ 1 IEX,D,Y,T,Z,P,S,time :  1  0.9980   0.000  -30.24  0.000  44.63   0.0000   0.0000
*******************************************************************************************
```

Excerpt from zgoubi.res : the final momentum resolution histogram at the spectrometer focal surface.

```
*******************************************************************************************
     16  HISTO       HISTO     OF

                    HISTOGRAMME  DE  LA  COORDONNEE  Y
                    PARTICULES  PRIMAIRES  ET  SECONDAIRES
                    DANS  LA  FENETRE :  -0.5000     /    2.000     (CM)
                    NORMALISE

   20
   19
   18
   17                                          Y
   16                                          Y         Y
   15                                          Y         Y
   14                                          Y         Y
   13                                          Y         Y
   12                                          Y         Y
   11                                          Y         Y
   10                                  0       0         0         0
    9                                  Y       Y         Y         Y
    8                                  Y       Y        YY         Y
    7                                  Y      YY        YY        YY
    6                     Y           YYY     YY        YY        YY
    5                     YY          YYY     YY        YY       YYYY
    4                     YY Y        YYY     YYY       YY       YYYY
    3                     YY YY      YYYYY    YYY      YYYY      YYYYY
    2                    YYYYYY     YYYYYY    YYYY     YYYYY     YYYYYY
    1                   YYYYYYYYYYYYYYYYYYYYYYYYYYYYYY YYYYYYYY   YYYYYYYY

          123456789012345678901234567890123456789012345678901234567890123456789012345678901
                   2         3         4         5         6         7         8         9


       TOTAL  COMPTAGE                :   10000  SUR  10000
       NUMERO   DU  CANAL  MOYEN      :      51
       COMPTAGE  AU   "      "        :     246
       VAL. PHYS. AU  "      "        :  0.750     (CM)
       RESOLUTION  PAR  CANAL         :  3.125E-02 (CM)

       PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                     COMPTAGE =  10000  PARTICULES
                     MIN = -0.1486    , MAX =   1.652    , MAX-MIN =   1.800     (CM)
                     MOYENNE =  0.7576     (CM)
                     SIGMA =  0.4621     (CM)

 TRAJ 1 IEX,D,Y,T,Z,P,S,time :  1  0.9980 0.2475 74.43 -6.2488E-03 -6.929   697.41 0.0000
*******************************************************************************************
```

2 TRANSFER MATRICES ALONG A TWO-STAGE SEPARATION KAON BEAM LINE

800 MeV/c kaon beam line at BNL Alternating Gradient Synchrotron [63]. The line includes two separation Wien filters.



Layout of LESB3 beamline.

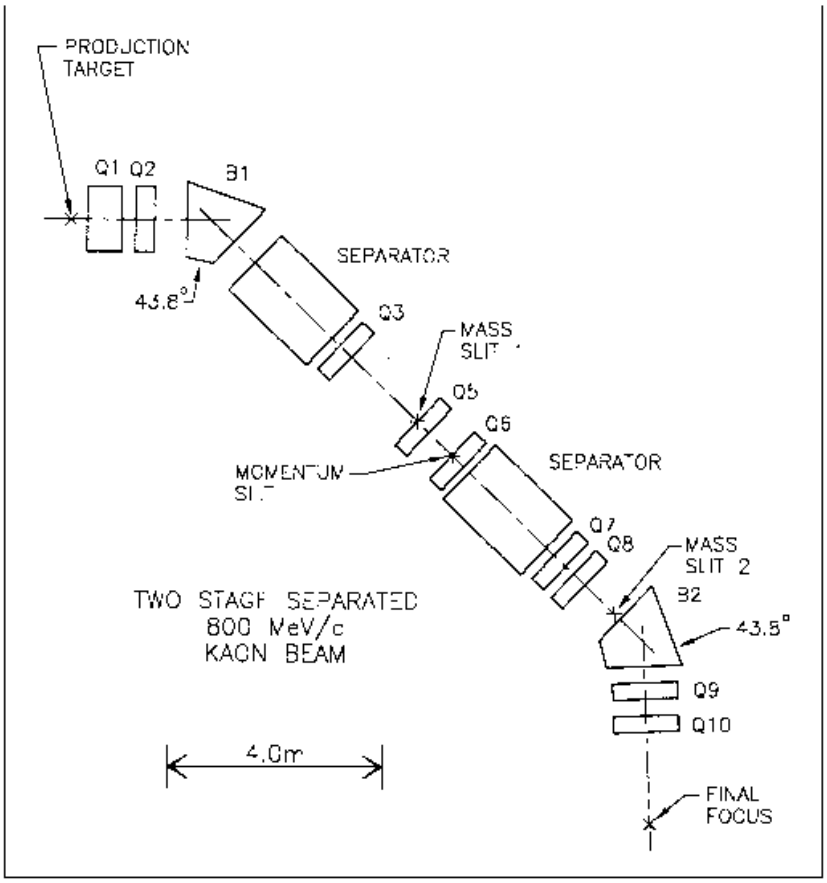


Figure 49: 800 MeV/c kaon beam line layout.

zgoubi data file.

```
 800 MeV/c KAON BEAM LINE. CALCULATION OF TRANSFER COEFFICIENTS.
'OBJET'                                                          1
2668.5100                          AUTOMATIC  GENERATION  OF
6                                  AN  OBJECT  FOR  CALCULATION
.1  .1  .1  .1 0.  .001            OF  THE  FIRST  ORDER  TRANSFER
0. 0. 0. 0. 1.                     COEFFICIENTS  WITH  'MATRIX'
'PARTICUL'                                                       2
 493.646 1.60217733E-19 0. 0. 0.   KAON M & Q, FOR USE IN WIEN FILTER
'DRIFT'                                                          3
 35.00000
'QUADRUPO'                           Q1                          4
0
  76.2  15.24  13.6
 30.  30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  1.1
 1  0. 0. 0.
'DRIFT'                                                          5
  25.00000
'QUADRUPO'                           Q2                          6
0
45.72  15.24   -11.357
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  1.1
 1  0. 0. 0.
'DRIFT'                                                          7
 -1.898
'TOSCA'                            3-D  MAP  THE  OF  FIRST      8
 0  0                                  BENDING  MAGNET
1.0313E-3  1. 1. 1.                B, X, Y, Z NORMALIZATION COEFFCNTS
1D map at z=0, from TOSCA
59  39  1 0
bw6_0.map
0  0. 0. 0.
2
  1.1
1 0 0 0
'CHANGREF'                                                       9
0.  -70.78  -43.8
'FAISCEAU'                                                      10
  'DRIFT'                                                       11
    -49.38
'OCTUPOLE'                                                      12
0
 10. 15.24  .6
 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 .4
 1  0. 0. 0.
'SEXTUPOL'                         SX1, COMPENSATION            13
0                                  OF  THE  Theta.Phi  ABERRATION
10.  15.24  2.4                    AT  VF1
 0.  0. 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 0. 0.  0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  .4
 1  0. 0. 0.
'DRIFT'                                                         14
  50.0
 'WIENFILT'                        FIRST  VERTICAL  WIEN  FILTER 15
 0
2.16  55.E5  -.0215576  2
20. 10. 10.
0.2401  1.8639  -0.5572  0.3904 0. 0.
0.2401  1.8639  -0.5572  0.3904 0. 0.
20. 10. 10.
0.2401  1.8639  -0.5572  0.3904 0. 0.
0.2401  1.8639  -0.5572  0.3904 0. 0.
1.
1. 0. 0. 0.
 'DRIFT'                                                        16
  30.
 'QUADRUPO'                          Q3                         17
0
  45.72  15.24  -6.34
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  1.1
 1  0. 0. 0.
'DRIFT'                                                         18
 10.0
'MULTIPOL'                         SX2 + OCTU, COMPENSATION     19
0                                  OF  THE  D.Phi  AND  D2.Phi
10.  15.24   0. 0. -8.  1.2  0. 0. 0. 0. 0. 0.    ABERRATIONS  AT  VF1
 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
  .4
 1  0. 0. 0.
'DRIFT'                                                         20
  90.0
'MATRIX'                           TRANSFER  COEFFICIENTS       21
2 0
'COLLIMA'                          FIRST  VERTICAL  FOCUS,      22
 2                                 MASS  SLIT
2  14.6  .15E10  0. 0.
'DRIFT'                                                         23
  20.0
'QUADRUPO'                           Q5                         24
0
 45.72  15.24   10.93
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
1.1
 1  0. 0. 0.
'DRIFT'                                                         25
  10.0
'MULTIPOL'                         COMPENSATION  OF            26
0                                  ABERRATIONS  AT  VF2
 10.  15.24  0. 0.  0.  1.  0. 0. 0. 0. 0. 0.
 0. 0.  0. 0. 0. 0. 0. 0. 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 0. 0.  0. 0. 0. 0. 0. 0. 0. 0.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 .4
 1  0. 0. 0.
'DRIFT'                                                         27
  10.0
'QUADRUPO'                           Q6                         28
0
 45.72  15.24  -11.18
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  1.1
 1  0. 0. 0.
'DRIFT'                                                         29
  50.0
 'WIENFILT'                        SECOND  VERTICAL  WIEN  FILTER 30
 0
2.16  -55.E5  .0215576  2
20. 10. 10.
0.2401  1.8639  -0.5572  0.3904 0. 0.
0.2401  1.8639  -0.5572  0.3904 0. 0.
20. 10. 10.
0.2401  1.8639  -0.5572  0.3904 0. 0.
0.2401  1.8639  -0.5572  0.3904 0. 0.
1.
1. 0. 0. 0.
'DRIFT'                                                         31
  30.0
'QUADRUPO'                           Q7                         32
0
 45.72  15.24  -6.44
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 1.1
 1  0. 0. 0.
'DRIFT'                                                         33
  25.00000
'QUADRUPO'                           Q8                         34
0
 45.72  15.24  8.085
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30. 30.
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  1.1
 1  0. 0. 0.
'DRIFT'                                                         35
  40.0
'COLLIMA'                          SECOND  VERTICAL  FOCUS,      36
 2                                 MASS  SLIT
1  17.   .2E10   0. 0.
'MATRIX'                           TRANSFER  COEFFICIENTS       37
2 0
'DRIFT'                                                         38
-25.0
'DIPOLE-M'                         SIMULATION  OF  THE  MAP     39
 3 0 0                             OF  THE  SECOND  BENDING  MAGNET
 150  60                           (UPGRADED VERSION OF 'AIMANT')
 18.999   0. 0. 0.
  79.3329   17.7656  140.4480   110.  170.
 15. -1.
 4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
 0.00    21.90  1.E6 -1.E6  1.E6  1.E6
 15. -1.
 4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
-43.80  -21.90 -1.E6 -1.E6  1.E6 -1.E6
 0. 0.
 4  .1455   2.2670  -.6395  1.1558  0. 0.  0.
-43.80  -21.90 -1.E6 -1.E6  1.E6 -1.E6 1E6
 0                                 NO SHIMMING
 2                                 SCD ORDER INTERPOLATION
  2.5
 2
 147.48099   -0.31007  147.48099    0.31007
'DRIFT'                                                         40
-15.00000
'QUADRUPO'                           Q9                         41
0
 35.56  12.7  -13.69  -13.91
 30.  25.4
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30.  25.4
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  .5
 1  0. 0. 0.
```

```
'DRIFT'                                                          42
  25.00000
'QUADRUPO'                          Q10                          43
0
  35.56  12.7  11.97
 30.  25.4
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
 30.  25.4
 4    0.2490   5.3630  -2.4100   0.9870   0.   0.
  1.1
 1  0. 0. 0.
'DRIFT'                                                          44
 200.0
'MATRIX'                            TRANSFER  COEFFICIENTS       45
2 0                                 AT  THE  FINAL  FOCUS
'END'                                                            46
```

Excerpt of zgoubi.res : first and second order transfer matrices and higher order coefficients at the end of the line.

```
            FIRST  ORDER  COEFFICIENTS  ( MKSA ):

        3.60453       -4.453265E-02   -3.049728E-04   -1.165832E-04     0.00000       -5.229783E-02
       -2.05368        0.270335        4.700517E-05    1.763910E-05     0.00000       -9.561918E-02
       2.240965E-05   -8.687757E-07    -3.60817       -1.731805E-02     0.00000       -7.815367E-02
       1.185290E-05   -4.356398E-07    -2.05043       -0.286991         0.00000       -3.983392E-02
      -0.387557        2.313953E-02   -2.264218E-05   -8.015244E-06     1.00000        0.374917
        0.00000         0.00000         0.00000         0.00000         0.00000         1.00000

       DetY-1 =      -0.1170246601,    DetZ-1 =       0.0000034613

       R12=0 at   0.1647     m,        R34=0 at  -0.6034E-01 m

   First order sympletic conditions (expected values = 0) :
     -0.1170        3.4614E-06   -1.8207E-04    3.0973E-05    4.6007E-04   -8.0561E-05

            SECOND  ORDER  COEFFICIENTS  ( MKSA ):

1 11   7.34       1 21  -1.78       1 31  1.399E-02   1 41  1.456E-02   1 51   0.00       1 61   36.3
1 12  -1.78       1 22  -530.       1 32 -1.308E-03   1 42 -1.743E-03   1 52   0.00       1 62   12.3
1 13  1.399E-02   1 23 -1.308E-03   1 33 -0.611       1 43 -0.522       1 53   0.00       1 63 -2.771E-02
1 14  1.456E-02   1 24 -1.743E-03   1 34 -0.522       1 44  0.163       1 54   0.00       1 64 -2.211E-02
1 15   0.00       1 25   0.00       1 35   0.00       1 45   0.00       1 55   0.00       1 65   0.00
1 16   36.3       1 26   12.3       1 36 -2.771E-02   1 46 -2.211E-02   1 56   0.00       1 66   2.88


2 11  -303.       2 21   3.81       2 31  3.684E-02   2 41  3.581E-02   2 51   0.00       2 61   144.
2 12   3.81       2 22  -62.9       2 32 -5.821E-04   2 42 -1.638E-04   2 52   0.00       2 62 -0.759
2 13  3.684E-02   2 23 -5.821E-04   2 33   1.05       2 43   1.94       2 53   0.00       2 63 -1.031E-02
2 14  3.581E-02   2 24 -1.638E-04   2 34   1.94       2 44   6.70       2 54   0.00       2 64 -4.285E-02
2 15   0.00       2 25   0.00       2 35   0.00       2 45   0.00       2 55   0.00       2 65   0.00
2 16   144.       2 26 -0.759       2 36 -1.031E-02   2 46 -4.285E-02   2 56   0.00       2 66  -65.3


3 11 -0.145       3 21  2.158E-02   3 31   20.6       3 41   86.0       3 51   0.00       3 61 -0.201
3 12  2.158E-02   3 22   64.6       3 32   1.61       3 42  0.496       3 52   0.00       3 62  8.793E-02
3 13   20.6       3 23   1.61       3 33  0.710       3 43  0.128       3 53   0.00       3 63   39.1
3 14   86.0       3 24  0.496       3 34  0.128       3 44   64.8       3 54   0.00       3 64   7.17
3 15   0.00       3 25   0.00       3 35   0.00       3 45   0.00       3 55   0.00       3 65   0.00
3 16 -0.201       3 26  8.793E-02   3 36   39.1       3 46   7.17       3 56   0.00       3 66   1.46


4 11 -8.254E-02   4 21  1.146E-02   4 31   10.7       4 41   47.3       4 51   0.00       4 61 -0.127
4 12  1.146E-02   4 22   33.0       4 32  0.787       4 42  0.157       4 52   0.00       4 62  3.566E-02
4 13   10.7       4 23  0.787       4 33  0.365       4 43  6.774E-02   4 53   0.00       4 63   17.5
4 14   47.3       4 24  0.157       4 34  6.774E-02   4 44   33.1       4 54   0.00       4 64   1.05
4 15   0.00       4 25   0.00       4 35   0.00       4 45   0.00       4 55   0.00       4 65   0.00
4 16 -0.127       4 26  3.566E-02   4 36   17.5       4 46   1.05       4 56   0.00       4 66  0.715


5 11   568.       5 21  -7.67       5 31 -5.970E-02   5 41 -5.682E-02   5 51   0.00       5 61  -251.
5 12  -7.67       5 22   225.       5 32  1.283E-03   5 42  6.947E-04   5 52   0.00       5 62   2.77
5 13 -5.970E-02   5 23  1.283E-03   5 33   19.2       5 43   10.2       5 53   0.00       5 63  0.215
5 14 -5.682E-02   5 24  6.947E-04   5 34   10.2       5 44   1.59       5 54   0.00       5 64  0.129
5 15   0.00       5 25   0.00       5 35   0.00       5 45   0.00       5 55   0.00       5 65   0.00
5 16  -251.       5 26   2.77       5 36  0.215       5 46  0.129       5 56   0.00       5 66   112.

            HIGHER  ORDER  COEFFICIENTS  ( MKSA ):

        Y/Y3            5784.8
        Y/T3           9.40037E+05
        Y/Z3           0.70673
        Y/P3           0.42104

        T/Y3           -18607.
        T/T3           1.04607E+05
        T/Z3          -0.10234
        T/P3           5.25793E-02

        Z/Y3            32.161
        Z/T3            18.425
        Z/Z3           -872.50
        Z/P3           -785.20

        P/Y3            15.460
        P/T3            7.5264
        P/Z3           -409.98
        P/P3           -389.15
```

3 IN-FLIGHT DECAY IN SPES 3

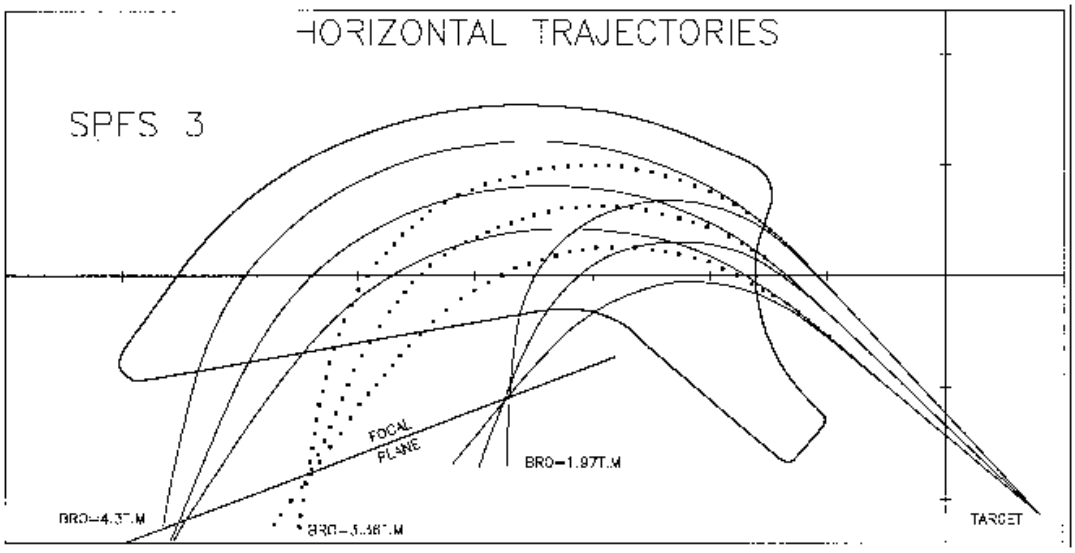


Figure 50: Layout of SPES 3 spectrometer.

zgoubi data file

```
SIMULATION  OF  PION  IN-FLIGHT  DECAY  IN  SPES3  SPECTROMETER
'MCOBJET'                                                          1
3360.                              REFERENCE  RIGIDITY (PION).
1                                  DISTRIBUTION  IN  WINDOW.
200                                BUNCHES  OF  200  PARTICLES.
1     1       1     1      1    1    UNIFORM DISTRIBUTION
0.    0.      0.    0.     0.   1.   CENTRAL  VALUES  OF  BARS.
.5e-2 50.e-3 .5e-2  50.e-3 0.  0.4   WIDTH  OF  BARS.
1     1       1     1      1    1    CUT-OFFS (UNUSED)
9   9. 9. 9. 9.                    UNUSED.
186387 548728 472874               SEEDS.
'PARTICUL'                                                         2
139.6000 0. 0. 26.03E-9 0.         PION MASS AND LIFE TIME
'MCDESINT'                                                         3
105.66   0.                        PION -> MUON + NEUTRINODECAY
136928 768370 548375
'ESL'                                                              4
77.3627
'CHAMBR'                           STOPS  ABERRANT  MUONS.         5
1
1   100. 10. 245.  0.
'DIPOLE'                                                           6
2
80.   208.5                        TOTAL ANGLE AT, CENTRAL RADIUS RM
33.   30. 0. 0. 0.                 CENTRAL ANGLE, FIELD, INDICES
46. -1.                            ENTRANCE EFB
4. .14552 5.21405 -3.38307 14.0629 0. 0. 0.
 15.   0.      -65.  0.    0.  -65.
46. -1.                            EXIT EFB
4. .14552 5.21405 -3.38307 14.0629 0. 0. 0.
-15.  69.       85.  0.    1.E6  1.E6
0. 0.                              LATERAL EFB (INHIBITED)
4. .14552 5.21405 -3.38307 14.0629 0. 0. 0.
-15.  69.       85.  0.    1.E6  1.E6 1E6
2 10.0                             2ND DEGREE INTERP., MESH 0.4 CM
4.                                 STEP SIZE
2
164.755 .479966 233.554 -.057963
'CHAMBR'                                                           7
2
1   100. 10. 245.  0.
'CHANGREF'                         TILT  ANGLE  OF                 8
0.   0.   -49.                     FOCAL  PLANE.
'HISTO'                            TOTAL  SPECTRUM (PION + MUON).  9
2  -170.  130.  60  1
20  'Y'  1  'Q'
'HISTO'                            PION  SPATIAL  SPECTRUM        10
2  -170.  130.  60  2              AT  FOCAL  PLANE.
20  'P'  1  'P'
'HISTO'                            MUON  SPATIAL  SPECTRUM        11
2  -170.  130.  60  3              AT  FOCAL  PLANE.
20  'y'  1  'S'
'HISTO'                            MUON  MOMENTUM  SPECTRUM       12
1    .2   1.7   60  3              AT  FOCAL  PLANE.
20  'd'  1  'S'
'REBELOTE'                         (49+1)  RUNS =  CALCULATION  OF 13
49  0.1  0                        (49+1)*200  TRAJECTORIES.
'END'                                                             14
```

Excerpt of zgoubi.res : histograms of primary and secondary particles at focal surface of SPES3.

```
********************************************************************************
     9  HISTO     TOTAL     SPECTRUM

                       HISTOGRAMME  DE  LA  COORDONNEE  Y
                       PARTICULES  PRIMAIRES  ET  SECONDAIRES
                       DANS  LA  FENETRE :  -1.7000E+02 /   1.3000E+02 (CM)
                       NORMALISE

  20
  19
  18
  17                                    Y     YY        Y  Y      Y
  16                                   Y YYYYY YY YYY  Y YY Y Y  YYYY
  15                                   Y YYYYY YYYYYYYYYYYYYY YYYYYYYY YY
  14                                  YYYYYYYYY YYYYYYYYYYYYYYYYYYYYYYYYY
  13                                  YYYYYYYYY YYYYYYYYYYYYYYYYYYYYYYYYY
  12                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
  11                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
  10                                  00000000000000000000000000000000000000
   9                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   8                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   7                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   6                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   5                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   4                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   3                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   2                                 YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
   1                                 YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY

                       123456789012345678901234567890123456789012345678901234567890
                               3         4         5         6         7         8


              TOTAL  COMPTAGE                :    9887  SUR  10000
              NUMERO   DU  CANAL  MOYEN      :      55
              COMPTAGE  AU    "      "       :     281
              VAL. PHYS. AU  "      "        :  0.000E+00 (CM)
              RESOLUTION  PAR  CANAL         :  5.000E+00 (CM)

              PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                      COMPTAGE =   9887  PARTICULES
                      MIN = -1.6687E+02, MAX = 9.4131E+01, MAX-MIN = 2.6100E+02(CM)
                      MOYENNE = -9.2496E-01 (CM)
                      SIGMA =  5.3583E+01 (CM)

********************************************************************************
     10  HISTO     PION      SPATIAL

                       HISTOGRAMME  DE  LA  COORDONNEE  Y
                       PARTICULES  PRIMAIRES
                       DANS  LA  FENETRE :  -1.7000E+02 /   1.3000E+02 (CM)
                       NORMALISE

  20
  19
  18                                       P
  17                                    P     PP        P  P      P
  16                                    PP    PP PPP  P PP P P  PPPPP
  15                                  P PPPPP PPPPPPPPPPPPPPPPPPPPPPP PP
  14                                 PPPPPPPPP PPPPPPPPPPPPPPPPPPPPPPPPPP
  13                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
  12                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
  11                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
  10                                 00000000000000000000000000000000000000
   9                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   8                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   7                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   6                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   5                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   4                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   3                                 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   2                                PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
   1                                PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP

                       123456789012345678901234567890123456789012345678901234567890
                               3         4         5         6         7         8


              TOTAL  COMPTAGE                :    9282  SUR  10000
              NUMERO   DU  CANAL  MOYEN      :      55
              COMPTAGE  AU    "      "       :     264
              VAL. PHYS. AU  "      "        :  0.000E+00 (CM)
              RESOLUTION  PAR  CANAL         :  5.000E+00 (CM)

              PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                      COMPTAGE =   9282  PARTICULES
                      MIN= -9.5838E+01, MAX = 9.3504E+01, MAX-MIN = 1.8934E+02 (CM)
                      MOYENNE =  4.9971E-01 (CM)
                      SIGMA =  5.3215E+01 (CM)

********************************************************************************
********************************************************************************
  11  HISTO     MUON      SPATIAL

                       HISTOGRAMME  DE  LA  COORDONNEE  Y
                       PARTICULES  SECONDAIRES
                       DANS  LA  FENETRE :  -1.7000E+02 /   1.3000E+02 (CM)
                       NORMALISE

20
19
18                                      y
17                                      y        yy
16                                      y        yy
15                                    y   y y     yyy   y
14                                    y   y y  y  yyyy  y
13                                    y   yyy  yy yyyy  y
12                                   yy y yyy yyy yyyyy y    y
11                                   yyy y yyy yyy yyyyy y    y   y
10                                   000 0 000 000000000000   0   00
 9                                   yyyyy yyy yyyyyyyyyyyy y y   yy
 8                                  yyyyyyyyyy yyyyyyyyyyyyyyy y   yy
 7                                 yyyyyyyyyyyyyyyyyyyyyyyyyyyy y  yyy
 6                                 yyyyyyyyyyyyyyyyyyyyyyyyyyyyy y yyyy
 5                               y  yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy yyy
 4                               y  yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
 3                               y  yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
 2                              yy   yy  yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
 1                              yy  yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy

                       1234567890123456789012345678901234567890123456789012345678901234567890
                               3         4         5         6         7         8


              TOTAL  COMPTAGE                :     605  SUR  10000
              NUMERO   DU  CANAL  MOYEN      :      50
              COMPTAGE  AU    "      "       :      14
              VAL. PHYS. AU  "      "        : -2.500E+01 (CM)
              RESOLUTION  PAR  CANAL         :  5.000E+00 (CM)

              PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                      COMPTAGE =    605  PARTICULES
                      MIN= -1.6687E+02, MAX = 9.4131E+01, MAX-MIN = 2.6100E+02 (CM)
                      MOYENNE = -2.2782E+01 (CM)
                      SIGMA =  5.4452E+01 (CM)

********************************************************************************
    12  HISTO     MUON      MOMENTUM

                       HISTOGRAMME  DE  LA  COORDONNEE    D
                       PARTICULES  SECONDAIRES
                       DANS  LA  FENETRE :   2.0000E-01 /   1.7000E+00
                       NORMALISE

20
19
18                                    d   d
17                                    d   d
16                                    d   dd
15                                   dd   dd
14                                  ddd d dd
13                                  ddd d dd     dd
12                                  ddd d dd     dd
11                                  ddd d ddd d  dd
10                                 0000 0 000 0  00
 9                                ddddd ddddd dd dd
 8                               ddddddddddddddd ddddd
 7                               ddddddddddddddddddddd
 6                              ddddddddddddddddddddddd ddd
 5                             ddddddddddddddddddddddddddddd
 4                           ddddddddddddddddddddddddddddddddd
 3                           ddddddddddddddddddddddddddddddddd  d
 2                          ddddddddddddddddddddddddddddddddddddd dddd
 1                          ddddddddddddddddddddddddddddddddddddddddddddd

                       12345678901234567890123456789012345678901234567890123456789
                               3         4         5         6         7


              TOTAL  COMPTAGE                :     605  SUR  10000
              NUMERO   DU  CANAL  MOYEN      :      46
              COMPTAGE  AU    "      "       :      16
              VAL. PHYS. AU  "      "        :  8.250E-01
              RESOLUTION  PAR  CANAL         :  2.500E-02

              PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                      COMPTAGE =    605  PARTICULES
                      MIN =  3.7184E-01, MAX =  1.3837E+00, MAX-MIN =  1.0119E+00
                      MOYENNE =  8.1693E-01
                      SIGMA =  2.2849E-01

********************************************************************************
```

4 USE OF THE FITTING PROCEDURE

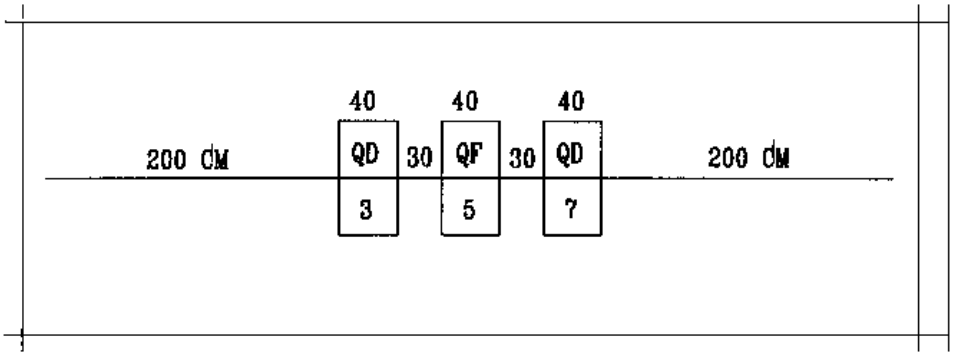


Figure 51: Vary B in all quadrupoles, for fitting of the transfer coefficients R_{12} and R_{34} at the end of the line. The first and last quadrupoles are coupled so as to present the same value of B.

zgoubi data file.

```
MATCHING A SYMMETRIC QUADRUPOLE TRIPLET
 'OBJET'                                                                    1
2501.73            RIGIDITY (kG.cm)
5                  11 PARTICLES GENERATED FOR USE OF MATRIX
2.  2.  2.   2.  0. .001
0.  0.  0.   0.  0.  1.
 'ESL   '                                                                   2
200.
 'QUADRUPO'    3                                                           3
0
40.  15.  -7.
0.  0.
6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
0.   0.
6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
5.
1 0. 0. 0.
 'ESL'                                                                      4
30.
 'QUADRUPO'     5                                                          5
0
40.  15.  3.
0.  0.
6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
0.   0.
6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
5.
1 0. 0. 0.
 'ESL'                                                                      6
30.
 'QUADRUPO'      7                                                         7
0
40.  15.  -7.
0.  0.
6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
0.   0.
6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
5.
1 0. 0. 0.
 'ESL'                                                                      8
200.
 'MATRIX'                                                                   9
1  0
 'FIT2'            VARY B IN QUADS FOR FIT OF R12 AND R34                  10
2                  # OF VARIABLES
3  12 7.12  2.     SYMMETRIC TRIPLET => QUADS #1 AND #3 ARE COUPLED
5  12  0.   2.     PRMTR #12 OF ELEMENTS #3, 5 AND 7 IS FIELD VALUE
2  1.E-10          # OF CONSTRAINTS, PENALTY
1  1 2 8 16.6 1.   CNSTRNT #1 : R12=16.6 AFTER LAST DRIFT (LMNT #8)
1  3 4 8 -.88 1.   CNSTRNT #2 : R34=-.88 AFTER LAST DRIFT
 'END'                                                                      11
```

Excerpt of zgoubi.res : first order transfer matrices prior to and after fitting.

```
************************************************************************************
 TRANSFER MATRIX WITH STARTING  CONDITIONS :


                 MATRICE  DE  TRANSFERT  ORDRE  1  ( MKSA )

    5.43427         17.0254         0.00000         0.00000         0.00000         0.00000
    1.67580         5.43425         0.00000         0.00000         0.00000         0.00000
    0.00000         0.00000        -1.27003        -0.974288        0.00000         0.00000
    0.00000         0.00000        -0.629171        -1.27004        0.00000         0.00000
    0.00000         0.00000         0.00000         0.00000         1.00000         0.00000
    0.00000         0.00000         0.00000         0.00000         0.00000         1.00000


************************************************************************************

STATE OF VARIABLES  AFTER  MATCHING :

 LMNT  VAR  PARAM  MINIMUM     INITIAL         FINAL         MAXIMUM      STEP     NAME       LBL1     LBL2
    3    1     12   -21.0       -7.00       -6.972765137       7.00      1.707E-04 QUADRUPO   3          *
    7    1    120   -6.97       -7.00       -6.972765137       7.00      1.707E-04
    5    2     12   -3.00        3.00        3.229344585       9.00      1.266E-04 QUADRUPO   5          *
 STATUS OF CONSTRAINTS (Target penalty =   1.0000E-10)
 TYPE  I   J  LMNT#       DESIRED           WEIGHT         REACHED         KI2          *  Parameter(s)
   1   1   2     8      1.6600000E+01     1.0000E+00    1.6599981E+01                  *  0 :
   1   3   4     8     -8.8000000E-01     1.0000E+00   -8.8000964E-01                  *  0 :
 Fit reached penalty value   8.4374E-11

************************************************************************************

MATRIX, WITH NEW VARIABLES :

    5.27056         16.6000         0.00000         0.00000         0.00000         0.00000
    1.61443         5.27450         0.00000         0.00000         0.00000         0.00000
    0.00000         0.00000        -1.24205        -0.880010        0.00000         0.00000
    0.00000         0.00000        -0.622553        -1.24620        0.00000         0.00000
    0.00000         0.00000         0.00000         0.00000         1.00000         0.00000
    0.00000         0.00000         0.00000         0.00000         0.00000         1.00000

************************************************************************************
```

5 MULTITURN SPIN TRACKING IN SATURNE 3 GeV SYNCHROTRON

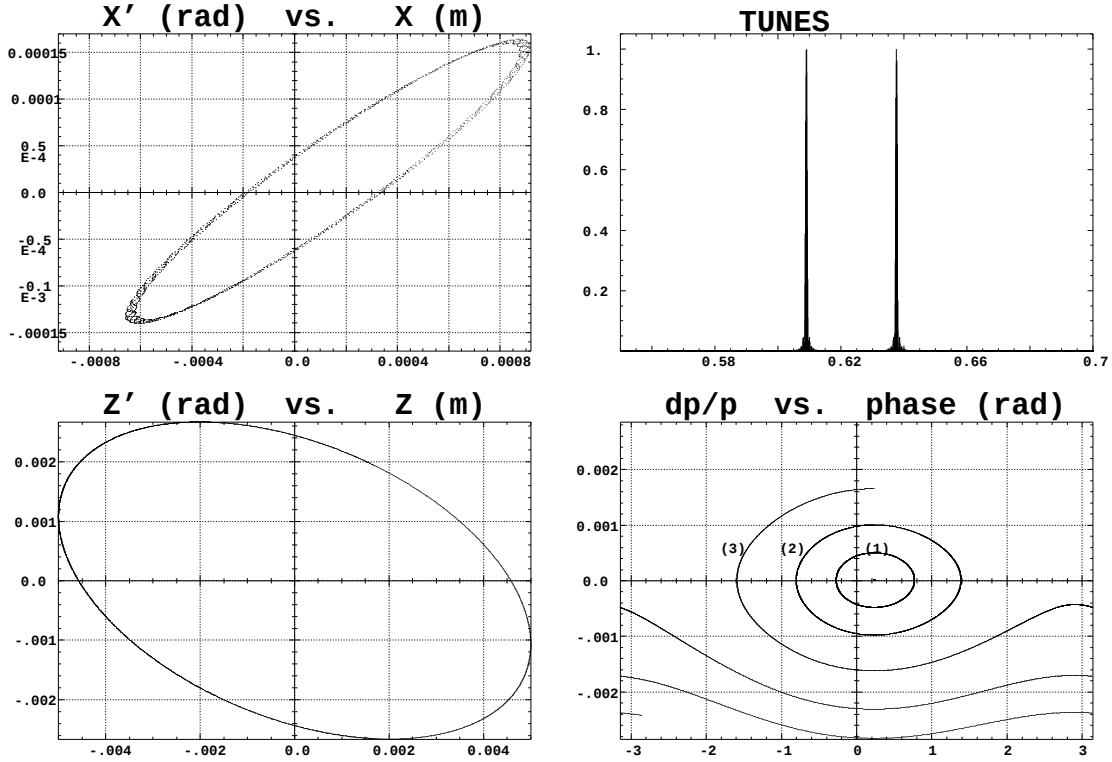


Figure 52: Tracking over 3000 turns. These simulations exhibit the first order parameters and motions as produced by the multi-turn ray-tracing.
(A) Horizontal phase-space: the particle has been launched near to the closed orbit (the fine structure is due to $Y - Z$ coupling induced by bends fringe fields, also responsible of the off-centering of the local closed orbit - at ellipse center).
(B) Vertical phase-space: the particle has been launched with $Z_0 = 4.58\ 10^{-3}$ m, $Z'_0 = 0$. A least-square fit by $\gamma_Z Z^2 + 2\alpha_Z Z Z' + \beta_Z Z'^2 = \varepsilon_Z/\pi$ yields $\beta_Z = 2.055$ m, $\alpha_Z = 0.444$, $\gamma_Z = 0.582$ m^{-1}, $\varepsilon_Z/\pi = 12\ 10^{-6}$ m.rad in agreement with matrix calculations.
(C) Fractional tune numbers obtained by Fourier analysis for $\varepsilon_Y/\pi = \varepsilon_Z/\pi \simeq 12\ 10^{-6}$ m.rad: $\nu_Y = 0.63795$, $\nu_Z = 0.60912$ (the integer part is 3 for both).
(D) Longitudinal phase-space ("$(DP, phase)$" in Zgoubi notations): particles with initial momentum dispersion of $5\ 10^{-4}$ (1), 10^{-3} (2), $1.65\ 10^{-3}$ (3) (out of acceptance), are accelerated at 1405 eV/turn ($\dot{B} = 2.1$ T/s); analytical calculations give accordingly momentum acceptance of $1.65\ 10^{-3}$.

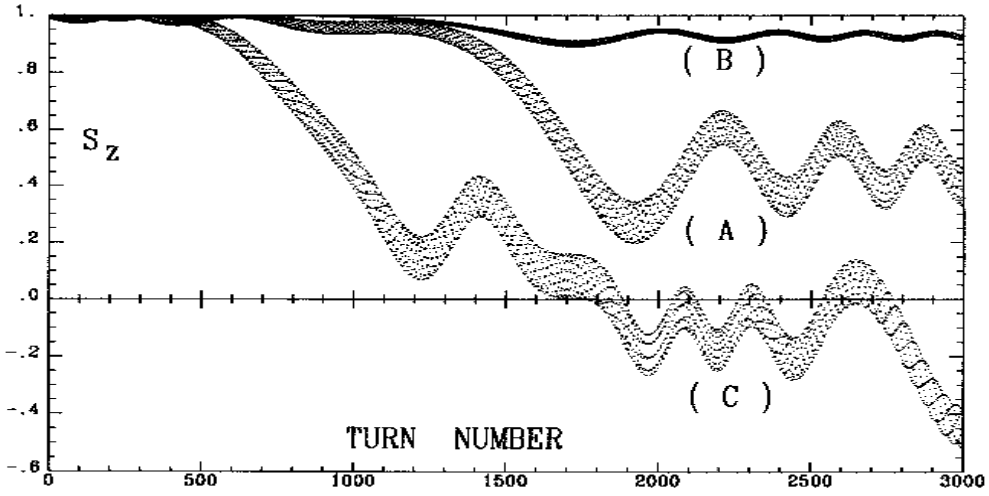


Figure 53: Crossing of $\gamma G = 7 - \nu_Z$, at $\dot{B} = 2.1$ T/s.
(A) $\varepsilon_Z/\pi = 12.2\ 10^{-6}$ m.rad. The strength of the resonance is $\mid \varepsilon \mid = 3.3\ 10^{-4}$. As expected from the Froissart-Stora formula [66] the asymptotic polarization is about 0.44.
(B) The emittance is now $\varepsilon_Z/\pi = 1.2\ 10^{-6}$ m.rad; comparison with (A) shows that $\mid \varepsilon \mid$ is proportional to $\sqrt{\varepsilon_Z}$.
(C) Crossing of this resonance for a particle having a momentum dispersion of 10^{-3}.

zgoubi data file (beginning and end).

```
 SATURNE. CROSSING GammaG=7-NUz, NUz=3.60877(perturbed)
 'OBJET'
 5015.388                                        834.04 MeV, proton
 2
  4  1
6.2E-02     6.5E-02   .458  0.  0.  1.00    'o'   EpsilonY/pi ~ 0. (Closed orbit)
0.356       0.379     .458  0.  0.  1.0005  '1'
0.647       0.689     .458  0.  0.  1.001   '2'
1.024       1.09      .458  0.  0.  1.0016  '3'
1 1 1 1
'SCALING'
1  4
  MULTIPOL
  2                                 CROSSING GammaG=7-Nuz+/-14E, E=3.3E-4
  5015.388E-3  5034.391E-3          AT 2.1 T/s, IN 3442 MACHINE TURNS,
  1            3442                 FROM 834.041 TO  838.877 MeV
  QUADRUPO
  2
  5015.388E-3  5034.391E-3
  1            3442
  BEND
  2
  5015.388E-3  5034.391E-3
  1            3442
  CAVITE
  2
  1.           1.00378894           RELATIVE CHANGE OF SYNCHRONOUS RIGIDITY
  1            3442
'PARTICUL'
 938.2723  1.6021892E-19  1.7928474 0. 0.
'SPNTRK'
3
'QUADRUPO'                QP  1                                          5
  0
 46.723  10.  .763695                 .763695  = FIELD FOR BORO=1 T.m
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 #30|50|30 Quad
 1 0. 0. 0.
 'ESL'                     SD  2                                           6
 71.6256
  'BEND'                   DIP 3 4 3                                      7
   0
  247.30039  0.  1.57776
    20. 8. .04276056667
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
    20. 8. .04276056667         20.  8.
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
  #30|120|30   bend    3 0. 0. 0. -.1963495408
 'ESL'                     SD  2                                           8
 71.6256
 'MULTIPOL'                QP  5                                          9
  0
 48.6273  10. 0. -.765533 0. 0. 0. 0.   0. 0. 0. 0.
 0.  0. 0. 0. 0. .0 .0  0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0. 0. 0. 0.  0. 0.   0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0. 0. 0. 0. 0. 0.   0. 0. 0. 0.
 #30|50|30 Quad
 1 0. 0. 0.
'ESL'                     SD  2                                          10
 71.6256
  'BEND'                   DIP 3 4 3                                     11
   0
  247.30039  0.  1.57776
    20. 8. .04276056667
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
    20. 8. .04276056667         20.  8.
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
  #30|120|30   bend    3 0. 0. 0. -.1963495408
 'ESL'                     SD  2                                          12
 71.6256
 'QUADRUPO'                QP  1                                          13
  0
 46.723  10.  .763695
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 #30|50|30 Quad
 1 0. 0. 0.
 'ESL'                     SD  2                                          14
 71.6256
  'BEND'                   DIP 3 4 3                                     15
   0
  247.30039  0.  1.57776
    20. 8. .04276056667
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
    20. 8. .04276056667         20.  8.
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
  #30|120|30   bend    3 0. 0. 0. -.1963495408
 'ESL'                     SD  2                                          16
 71.6256
 'MULTIPOL'                QP  5                                          17
  0
 48.6273  10.  0.  -.765533 0. 0. 0. 0.   0. 0. 0. 0.
 0.  0. 0. 0. 0.  0. 0.    0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0. 0. 0. 0.  0. 0.    0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0. 0. 0. 0. 0. 0.   0. 0. 0. 0.
 #30|50|30 Quad
 1 0. 0. 0.
 'ESL'                     SD  2                                          18
 71.6256
  'BEND'                   DIP 3 4 3                                     19
   0
  247.30039  0.  1.57776
    20. 8. .04276056667
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
    20. 8. .04276056667         20.  8.
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
  #30|120|30   bend    3 0. 0. 0. -.1963495408
 'ESL'                     SD  2                                          20
 71.6256
 'QUADRUPO'                QP  1                                          21
  0
 46.723  10.  .763695
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 #30|50|30 Quad
 1 0. 0. 0.
'ESL'                                                                    22
392.148
 'MULTIPOL'                QP  5                                          23
  0
 48.6273  10.  0.  -.765533 0. 0. 0. 0.   0. 0. 0. 0.
 0.  0. 0. 0. 0.  0. 0.   0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0. 0. 0. 0.  0. 0.   0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0. 0. 0. 0. 0. 0.    0. 0. 0. 0.
 #30|50|30 Quad
 1 0. 0. 0.
'ESL'                                                                    24
392.148
 'QUADRUPO'                QP  1                                          25
  0
 46.723  10.  .763695
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 #30|50|30 Quad
 1 0. 0. 0.
 'ESL'                     SD  2                                          26
 71.6256
  'BEND'                   DIP 3 4 3                                     27
   0
  247.30039  0.  1.57776
    20. 8. .04276056667
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
    20. 8. .04276056667         20.  8.
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
  #30|120|30   bend    3 0. 0. 0. -.1963495408
 'ESL'                     SD  2                                          28
 71.6256
 'MULTIPOL'                QP  5                                          29
  0
 48.6273  10.  0. -.765533  0. 0. 0. 0.   0. 0. 0. 0.
 0.  0. 0. 0. 0.  0. 0.    0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0.  0. 0. 0. 0.  0. 0.   0. 0. 0. 0.
 6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
 0. 0. 0. 0. 0. 0.    0. 0. 0. 0.
 #30|50|30 Quad
 1 0. 0. 0.
'ESL'                     SD  2                                          30
 71.6256
  'BEND'                   DIP 3 4 3                                     31
   0
  247.30039  0.  1.57776
    20. 8. .04276056667
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
    20. 8. .04276056667         20.  8.
  4 .2401  1.8639  -.5572  .3904 0. 0. 0.
  #30|120|30   bend    3 0. 0. 0. -.1963495408

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

 'ESL'                                                                    84
 392.148
 'CAVITE'                                                                 85
  1
 105.5556848673   3.
 6000.      0.                    SIN(phis) = .234162, dE=1.40497 keV/Turn.
'FAISCNL'                                                                 86
 b_zgoubi.fai
 'SPNPRNL'                                                                87
 zgoubi.spn
 'SPNPRT'                                                                 88
 'REBELOTE'                                                               90
 2999  0.1 99                      TOTAL NUMBER OF TURNS = 3000
'END'                                                                     91
```

6 MICRO-BEAM FOCUSING WITH $\vec{E} \times \vec{B}$ QUADRUPOLES

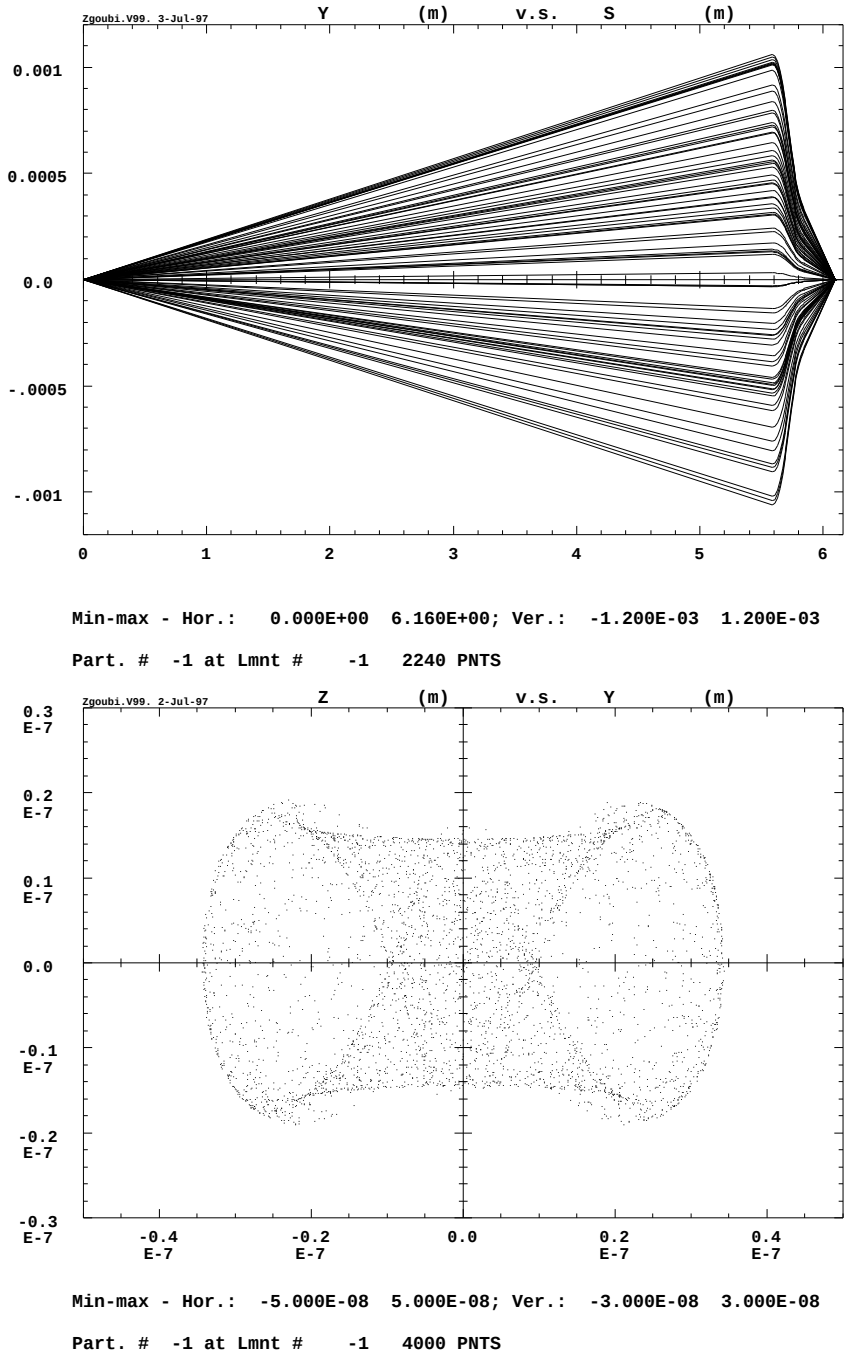


Figure 54: *Upper plot*: 50-particle beam tube ray-traced through a double focusing quadrupole doublet typical of the front end design of micro-beam lines. Initial conditions are : $Y_0 = Z_0 = 0$, angles T_0 and P_0 random uniform within ± 0.2 mrad, and momentum dispersion $\delta p/p$ uniform in $\pm 3\,10^{-4}$.
Lower plot: **(D)** sub-micronic cross-section at the image plane of a 4000-particle beam with initial conditions as above, obtained thanks to the second-order achromatic magneto-electrostatic quadrupole doublet (the image size would be $\Delta Y \approx \Delta Z \approx \pm 50 \mu m$ with regular magnetic quadrupoles, due to the momentum dispersion). Note the high resolution of the ray-tracing which still reveals image structure of nanometric size.

zgoubi data file.

```
  MICROBEAM LINE, WITH A MAGNETO-ELECTROSTATIC QUADRUPOLE DOUBLET.
  'MCOBJET'                                 RANDOM  OBJECT  DEFINITION      1
  20.435                                    RIGIDITY (20keV  PROTONS).
  1                                         DISTRIBUTION IN WINDOW.
  200                                       NUMBER  OF  PARTICLES.
  1    1     1    1     1    1              UNIFORM DISTRIBUTION.
  0.   0.    0.   0.    0.   1.             CENTRAL  VALUE,  AND
  0.  .2e-3  0.  .2e-3  0.   0.0003         HALF WIDTH OF DISTRIBUTION.
  10.  10.   10.  10.   10.  10.            CUT-OFFS (UNUSED).
  9   9. 9. 9. 9.                           FOR P(D) - UNUSED.
  186387 548728 472874                      SEEDS.
   'PARTICUL'                               PARTICLE MASS AND CHARGE        2
    938.2723 1.60217733E-19 0. 0. 0.          FOR INTEGRATION IN E-FIELD.
  'DRIFT'                                   DRIFT.                          3
  500.
  'DRIFT'                                   DRIFT.                          4
  59.
  'EBMULT'                                  FIRST MAGNETO-ELECTROSTATIC     5
   0                                                 QUADRUPOLE.
  10.2  10. 0. -9272.986  0. 0. 0. 0. 0. 0. 0. 0.     ELECTRIC Q-POLE COMPONENT.
   0.  0.  0.  0.  0.  0.  0.  0. 0. 0. 0.            ENTRANCE EFB, SHARP EDGE.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
   0.  0.  0.  0.  0.  0.  0.  0. 0. 0. 0.            EXIT EFB, SHARP EDGE.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
   0. 0. 0. 0. 0. 0. 0. 0. 0.
  10.2  10. 0.  1.89493  0. 0. 0. 0.  0. 0. 0. 0.     MAGNETIC Q-POLE COMPONENT.
   0.  0.  0.  0.  0.  0.  0.  0. 0. 0. 0.            ENTRANCE EFB, SHARP EDGE.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
   0.  0.  0.  0.  0.  0.  0.   0. 0. 0. 0.           EXIT EFB, SHARP EDGE.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
  0.  0. 0. 0. 0. 0. 0. 0. 0. 0.
  .8
  1 0. 0. 0.
  'DRIFT'                                   DRIFT.                          6
  4.9
  'EBMULT'                                  SECOND MAGNETO-ELECTROSTATIC    7
   0                                                 QUADRUPOLE.
  10.2  10. 0.  13779.90  0. 0. 0. 0. 0. 0. 0. 0.
   0.  0.  0.  0.  0.  0.  0. 0. 0. 0. 0.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
   0.  0.  0.  0.  0.  0.  0. 0. 0. 0. 0.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
  0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
  10.2  10. 0.   -2.81592  0. 0. 0. 0. 0. 0. 0. 0.
   0.  0.  0.  0.  0.  0.  0. 0. 0. 0. 0.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
   0.  0.  0.  0.  0.  0.  0. 0. 0. 0. 0.
  6  .1122 6.2671 -1.4982 3.5882 -2.1209 1.723
  0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
  .8
  1 0. 0. 0.
  'DRIFT'                                   DRIFT.                          8
  25.
  'HISTO'                                   HISTOGRAM                       9
  2  -5E-6  5E-6     60  2                  OF THE Y COORDINATE.
  20  'Y'  1  'Q'
  'HISTO'                                   HISTOGRAM                      10
  4  -5E-6  5E-6     60  2                  OF THE Z COORDINATE.
  20  'Z'  1  'Q'
  'FAISCNL'                        RAYS ARE STORED IN rays.out             11
  rays.out                         FOR FURTHER PLOTTING.
  'REBELOTE'                       RUN AGAIN, FOR RAY-TRACING              12
  19  0.1 0                        A TOTAL OF 200*(19+1) PARTICLES.
  'END'                                                                    13
```

zgoubi.res file.

```
****************************************************************************************************************************************************
     12  REBELOTE    RUN         AGAIN

     Multiple pass,
          from element #     1 : MCOBJET   /label1=RANDOM    /label2=OBJECT     to REBELOTE /label1=RANDOM    /label2=OBJECT
          ending at pass #      20 at element #    12 : REBELOTE  /label1=RUN       /label2=AGAIN

     End of pass #       19 through the optical structure

                     Total of       3800 particles have been launched

****************************************************************************************************************************************************

      Next  pass  is  #    20 and  last  pass  through  the  optical  structure

****************************************************************************************************************************************************
      1  MCOBJET     RANDOM      OBJECT
                          Reference  magnetic rigidity =         20.435 KG*CM

                          Object  built  up  of     200  particles
                     Distribution in a Window
                     Central values (MKSA units):
               Yo, To, Zo, Po, Xo, BR/BORO  :          0.000       0.000       0.000       0.000       0.000       1.000
                     Width  (  +/- , MKSA units ) :
               DY, DT, DZ, DP, DX, DBR/BORO :           0.00       2.000E-04    0.00       2.000E-04    0.00       3.000E-04
                     Cut-offs  ( * +/-Width ) :
               NY, NT, NZ, NP, NX, NBR/BORO :               1.00        1.00        1.00        1.00        1.00        1.00
                     Type  of  sorting :
               Y, T, Z, P, X, D :                      Uniform     Uniform     Uniform     Uniform     Uniform     Uniform

****************************************************************************************************************************************************
      2  PARTICUL  PARTICLE  MASS

     Particle  properties :

                          Mass          =    938.272        MeV/c2
                          Charge        =   1.602177E-19    C

                Reference  data :
                     mag. rigidity (kG.cm)   :   20.435000      =p/q, such that dev.=B*L/rigidity
                     mass (MeV/c2)           :   938.27230
                     momentum (MeV/c)        :   6.1262621
                     energy, total (MeV)     :   938.29230
                     energy, kinetic (MeV)   :  1.99998909E-02
                     beta = v/c              :  6.5291616518E-03
                     gamma                   :   1.000021316
                     beta*gamma              :  6.5293008252E-03
                     electric rigidity (MeV) :  3.9999376635E-02  =T[eV]*(gamma+1)/gamma, such that dev.=E*L/rigidity

****************************************************************************************************************************************************
```

```
**********************************************************************************************************************************
      3  DRIFT       DRIFT.      3

                            Drift,  length =   500.00000  cm

TRAJ #1 IEX,D,Y,T,Z,P,S,time :  1  2.999865E-04 -2.027052E-02 -4.054104E-02 -9.359114E-02 -1.871823E-01   5.0000001E+02   2.55365E+00

 Cumulative length of optical axis =    5.00000000     m  at  ;  Time  (for ref. rigidity & particle) =   2.554417E-06 s

**********************************************************************************************************************************
      4  DRIFT       DRIFT.      4

                            Drift,  length =    59.00000  cm

TRAJ #1 IEX,D,Y,T,Z,P,S,time :  1  2.999865E-04 -2.266244E-02 -4.054104E-02 -1.046349E-01 -1.871823E-01   5.5900001E+02   2.85498E+00

 Cumulative length of optical axis =    5.59000000     m  at  ;  Time  (for ref. rigidity & particle) =   2.855839E-06 s

**********************************************************************************************************************************
      5  EBMULT      FIRST       ELECTRO-MA

      -----  MULTIPOLE   :
               Length  of  element  =    10.200000      cm
               Bore  radius      RO =    10.000      cm
              E-DIPOLE      =  0.0000000E+00 V/m
              E-QUADRUPOLE  = -9.2729860E+03 V/m
              E-SEXTUPOLE   =  0.0000000E+00 V/m
              E-OCTUPOLE    =  0.0000000E+00 V/m
              E-DECAPOLE    =  0.0000000E+00 V/m
              E-DODECAPOLE  =  0.0000000E+00 V/m
              E-14-POLE     =  0.0000000E+00 V/m
              E-16-POLE     =  0.0000000E+00 V/m
              E-18-POLE     =  0.0000000E+00 V/m
              E-20-POLE     =  0.0000000E+00 V/m

              Entrance/exit field models are sharp edge
              FINTE, FINTS, gap :    0.0000E+00   0.0000E+00   5.0000E+00

      -----  MULTIPOLE   :
               Length  of  element  =    10.200000      cm
               Bore  radius      RO =    10.000      cm
              B-DIPOLE      =  0.0000000E+00 kG
              B-QUADRUPOLE  =  1.8949300E+00 kG
              B-SEXTUPOLE   =  0.0000000E+00 kG
              B-OCTUPOLE    =  0.0000000E+00 kG
              B-DECAPOLE    =  0.0000000E+00 kG
              B-DODECAPOLE  =  0.0000000E+00 kG
              B-14-POLE     =  0.0000000E+00 kG
              B-16-POLE     =  0.0000000E+00 kG
              B-18-POLE     =  0.0000000E+00 kG
              B-20-POLE     =  0.0000000E+00 kG

              Entrance/exit field models are sharp edge
              FINTE, FINTS, gap :    0.0000E+00   0.0000E+00   5.0000E+00

  ***  Warning : sharp edge model, vertical wedge focusing approximated with first order kick. FINT at entrance =    0.000

  ***  Warning : sharp edge model,  vertical wedge focusing approximated with first order kick. FINT at exit =    0.000

                   Integration step :  0.8000     cm

 Cumulative length of optical axis =    5.69200000     m ;  Time  (for ref. rigidity & particle) =   2.907949E-06 s

**********************************************************************************************************************************
      6  DRIFT       DRIFT.      6

                            Drift,  length =     4.90000  cm

TRAJ #1 IEX,D,Y,T,Z,P,S,time :  1  2.839285E-04 -1.310701E-02  9.564949E-01 -1.603604E-01 -5.590035E+00   5.7410014E+02   2.93210E+00

 Cumulative length of optical axis =    5.74100000     m  at  ;  Time  (for ref. rigidity & particle) =   2.932982E-06 s

**********************************************************************************************************************************
      7  EBMULT      SECOND      ELECTRO-MA

      -----  MULTIPOLE   :
               Length  of  element  =    10.200000      cm
               Bore  radius      RO =    10.000      cm
              E-DIPOLE      =  0.0000000E+00 V/m
              E-QUADRUPOLE  =  1.3779900E+04 V/m
              E-SEXTUPOLE   =  0.0000000E+00 V/m
              E-OCTUPOLE    =  0.0000000E+00 V/m
              E-DECAPOLE    =  0.0000000E+00 V/m
              E-DODECAPOLE  =  0.0000000E+00 V/m
              E-14-POLE     =  0.0000000E+00 V/m
              E-16-POLE     =  0.0000000E+00 V/m
              E-18-POLE     =  0.0000000E+00 V/m
              E-20-POLE     =  0.0000000E+00 V/m

              Entrance/exit field models are sharp edge
              FINTE, FINTS, gap :    0.0000E+00   0.0000E+00   5.0000E+00

      -----  MULTIPOLE   :
               Length  of  element  =    10.200000      cm
               Bore  radius      RO =    10.000      cm
              B-DIPOLE      =  0.0000000E+00 kG
              B-QUADRUPOLE  = -2.8159200E+00 kG
              B-SEXTUPOLE   =  0.0000000E+00 kG
              B-OCTUPOLE    =  0.0000000E+00 kG
              B-DECAPOLE    =  0.0000000E+00 kG
              B-DODECAPOLE  =  0.0000000E+00 kG
              B-14-POLE     =  0.0000000E+00 kG
              B-16-POLE     =  0.0000000E+00 kG
              B-18-POLE     =  0.0000000E+00 kG
              B-20-POLE     =  0.0000000E+00 kG

              Entrance/exit field models are sharp edge
              FINTE, FINTS, gap :    0.0000E+00   0.0000E+00   5.0000E+00

  ***  Warning : sharp edge model, vertical wedge focusing approximated with first order kick. FINT at entrance =    0.000

  ***  Warning : sharp edge model,  vertical wedge focusing approximated with first order kick. FINT at exit =    0.000

                   Integration step :  0.8000     cm

 Cumulative length of optical axis =    5.84300000     m ;  Time  (for ref. rigidity & particle) =   2.985092E-06 s

**********************************************************************************************************************************
```

```
     9  HISTO     HISTOGRA

                          HISTOGRAMME  DE  LA  COORDONNEE  Y
                          PARTICULES  PRIMAIRES  ET  SECONDAIRES
                          DANS  LA  FENETRE :  -5.0000E-06 /   5.0000E-06 (CM)
                          NORMALISE

   20
   19
   18                                              Y
   17                                              Y
   16                                              Y
   15                                              Y
   14                                            YYY     Y
   13                                          Y  YYY    YY YY
   12                                          YYYYYY YYYYYYYY             Y
   11                                  Y  Y      Y YYYYYYYYYYYYYYYYY  YY    Y  Y
   10                                  0  0   00000000000000000000000 00    0  0
    9                                  Y YYYY YYYYYYYYYYYYYYYYYYYYYYYY    Y YY
    8                                  Y YYYY YYYYYYYYYYYYYYYYYYYYYYYYY  Y YY
    7                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
    6                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
    5                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
    4                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
    3                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
    2                                  YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
    1                                 YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY

                          12345678901234567890123456789012345678901234567890123456789012345678901
                                    3         4         5         6         7         8

               TOTAL  COMPTAGE                :    4000  SUR   4000
               NUMERO   DU  CANAL  MOYEN      :      51
               COMPTAGE  AU   "      "         :     109
               VAL. PHYS. AU  "      "         :  0.000E+00 (CM)
               RESOLUTION  PAR  CANAL          :  1.667E-07 (CM)

               PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                          COMPTAGE =   4000  PARTICULES
                          MIN = -3.4326E-06, MAX =  3.4347E-06, MAX-MIN =  6.8674E-06 (CM)
                          MOYENNE = -2.8531E-08 (CM)
                          SIGMA =  1.8619E-06 (CM)

  TRAJ #1  D,Y,T,Z,P,S,IEX :  1.0002E+00  9.0257E-07 -2.3996E-01 -1.0770E-06  1.7947E+00      6.09300E+02  1
********************************************************************************************************************************************
     10  HISTO     HISTOGRA

                          HISTOGRAMME  DE  LA  COORDONNEE  Z
                          PARTICULES  PRIMAIRES  ET  SECONDAIRES
                          DANS  LA  FENETRE :  -5.0000E-06 /   5.0000E-06 (CM)
                          NORMALISE

   20
   19
   18
   17                                                    Z
   16                                        Z           Z
   15                                        ZZ          Z
   14                                        ZZ          Z
   13                                        ZZ         ZZ
   12                                        ZZ          ZZ
   11                                        ZZ         ZZZ
   10                                        00         000
    9                                        ZZZZ       ZZZ
    8                                       ZZZZZZZ Z Z   ZZZZZZ
    7                                       ZZZZZZZ Z Z ZZZZZZZZZ
    6                                       ZZZZZZZZZZZZZZZZZZZZ
    5                                       ZZZZZZZZZZZZZZZZZZZZ
    4                                       ZZZZZZZZZZZZZZZZZZZZ
    3                                      ZZZZZZZZZZZZZZZZZZZZZZ
    2                                      ZZZZZZZZZZZZZZZZZZZZZZ
    1                                      ZZZZZZZZZZZZZZZZZZZZZZ

                          12345678901234567890123456789012345678901234567890123456789012345678901
                                    3         4         5         6         7         8

               TOTAL  COMPTAGE                :    4000  SUR   4000
               NUMERO   DU  CANAL  MOYEN      :      51
               COMPTAGE  AU   "      "         :     169
               VAL. PHYS. AU  "      "         :  0.000E+00 (CM)
               RESOLUTION  PAR  CANAL          :  1.667E-07 (CM)

               PARAMETRES  PHYSIQUES  DE  LA  DISTRIBUTION :
                          COMPTAGE =   4000  PARTICULES
                          MIN = -1.9150E-06, MAX =  1.9110E-06, MAX-MIN =  3.8260E-06 (CM)
                          MOYENNE = -3.8539E-09 (CM)
                          SIGMA =  1.1232E-06 (CM)

  TRAJ #1  D,Y,T,Z,P,S,IEX :  1.0002E+00  9.0257E-07 -2.3996E-01 -1.0770E-06  1.7947E+00      6.09300E+02  1
********************************************************************************************************************************************
     11  FAISCNL   RAYS      ARE
  Print[s] occur at

********************************************************************************************************************************************
     12  REBELOTE  RUN       AGAIN,

                        ****  FIN  D'EFFET  DE  'REBELOTE'  ****
      IL  Y  A  EU        20  PASSAGES  DANS  LA  STRUCTURE
                    # PARTICULES  ENVOYEES      :      4000

********************************************************************************************************************************************
           PGM PRINCIPAL : ARRET SUR CLE   REBELOTE
********************************************************************************************************************************************
```

PART D

Running zgoubi and its post-processor/graphic interface zpop

INTRODUCTION

The **zgoubi** package, including this guide, examples, the **zpop** graphic/analysis post-processor, is available on web [5].
The **zgoubi** *FORTRAN* package is transportable; it has been compiled, linked and executed over the years on several types of systems (e.g. CDC, CRAY, IBM, DEC, HP, SUN, VAX, UNIX, LINUX).

An additional *FORTRAN* code, **zpop**, allows the post-processing and graphic treatment of **zgoubi** output files. **zpop** has been routinely used on DEC, HP, SUN stations, and is now on UNIX and LINUX systems.

1 GETTING TO RUN zgoubi AND zpop

1.1 Making the Executable Files zgoubi and zpop

1.1.1 The transportable package zgoubi

Compile and link the *FORTRAN* source files (normally, just run the Makefile), to create the executable **zgoubi**.

zgoubi is written in standard *FORTRAN*, mostly 77, therefore it does not require linking to any library.

1.1.2 The post-processor and graphic interface package zpop

Compile the *FORTRAN* source files (normally, just run the Makefile).

Link **zpop** with the graphic library, libminigraf.a [67]. This will create the executable **zpop**, that is to be run on an xterm type window.

1.2 Running zgoubi

The principles are the following:

- Fill up zgoubi.dat with the input data that describe the problem (see examples, Part C).
- Run **zgoubi** - no argument needed, default input is zgoubi.dat.
- Results of the execution will be printed into zgoubi.res and, upon options appearing in zgoubi.dat, into several other outputs files (see section 2 below).
- The command **zgoubi** acccepts the following arguments (see zgobui_main.f source file) :

  -fileIn, followed by an input file name,

  -fileOut, followed by an output file name,

  -fileLog, followed by a log file name,

  -saveExec, this will cause saving a copy of **zgoubi** exec in the local directory,

  -saveZpop, this will cause saving a copy of **zpop** exec in the local directory.

1.3 Running zpop

- Run **zpop** on an xterm window. This will open a graphic window.
- To access the data plotting sub-menu, select option 7.
- To access the data treatment sub-menu, select option 8.
- An on-line Help provides some information regarding the available post-processing procedures (Fourier transform, elliptical fit, synchrotron radiation, field map contours, tune diagram, etc.).

2 STORAGE FILES

When explicitly requested by means of the adequate keywords, options, or dedicated *LABEL*'s, extra storage files are opened by **zgoubi** (*FORTRAN "OPEN"* statement) and filled.

Their content can be post-processed using the interactive program **zpop** and its dedicated graphic and analysis procedures.

Most of these files are formmatted in columns, plotting their content using 'gnuplot' [71] is straightforward. A library of 'gnuplot' command files can be found in **zgoubi** sourceForge repository [5].

The **zgoubi** procedures that create and fill these extra output files are the following (refer to Part A and Part B of the guide):

- Keywords *FAISCNL, FAISTORE*: fill a '.fai' type file (normally named [b_]zgoubi.fai) with particle and spin coordinates and a lot of other informations.
- Keywords *SPNPRNL, SPNSTORE*: fill a '.spn' type file (normally named [b_]zgoubi.spn) with spin coordinates and other informations.
- Option $IC = 2$, with field map keywords (e.g. *CARTEMES*, *TOSCA*) : fill zgoubi.map with 2-D field map.
- Option $IL = 2$, with magnetic and electric element keywords: fill zgoubi.plt with the particle coordinates and experienced fields, step after step, all along the optical element.
- Using the keyword *MARKER* with '.plt' as a second *LABEL* will cause storage of current coordinates into zgoubi.plt.

Typical examples of graphics that one can expect from the post-processing of these files by **zpop** are the following (see examples, Part C):

- '.fai' type files

  Phase-space plots (transverse and longitudinal), aberration curves, at the position where *FAISTORE* or *FAISCNL* appears in the optical structure. Histograms of coordinates. Fourier analysis (e.g. , spin motion, tunes, in case of multi-turn tracking), calculation of Twiss parameters from phase-space ellipse matching.

- zgoubi.map

  Isomagnetic field lines of 2-D map. Superimposing trajectories read from zgoubi.plt is possible.

- zgoubi.plt

  Trajectories inside magnets and other lenses (these can be superimposed over field lines obtained from zgoubi.map). Fields experienced by the particles at the traversal of optical elements. Spectral and angular distributions of synchrotron radiation.

- zgoubi.spn

  Spin coordinates and histograms, at the position where *SPNPRNL* appears in the structure. Resonance crossing when performing multi-turn tracking.

In addition, many routines will deliver execution outcomes in individual output files if requested, usually by initroducing a '*PRINT*' command under the keyword of concern. Instances are zgoubi.MATRIX.out (keyword *MATRIX*), zgoubi.TWISS.out (keyword *OPTICS*), zgoubi.PICKUP.out (keyword *PICKUPS*), zgoubi.TWISS.out (keyword *TWISS*), etc.

References

[1] F. Méot et S. Valéro, *Manuel d'utilisation de Zgoubi*, Rapport IRF/LNS/88-13, CEA Saclay, 1988.

[2] F. Méot and S. Valéro, *Zgoubi users' guide*, Int. Rep. CEA/DSM/LNS/GT/90-05, CEA Saclay (1990) & TRIUMF report TRI/CD/90-02 (1990).

[3] F. Méot and S. Valéro, *Zgoubi users' guide - Version 3*, Int. Rep. DSM/LNS/GT/93-12, CEA Saclay (1993).

[4] F. Méot and S. Valero, *Zgoubi users' guide - Version 4*, FNAL Tech. Rep. FERMILAB-TM-2010 (Aug. 1997), & Int. Rep. CEA DSM DAPNIA/SEA-97-13, Saclay (Oct. 1997).

[5] http://zgoubi.sourceforge.net/

[6] F. Méot, *The electrification of Zgoubi*, SATURNE report DSM/LNS/GT/93-09, CEA Saclay (1993) ;
F. Méot, *Generalization of the Zgoubi method for ray-tracing to include electric fields*, NIM A 340 (1994) 594-604.

[7] D. Carvounas, *Suivi numérique de particules chargées dans un solénoïde*, rapport de stage, CEA/LNS/GT, 1991.

[8] F. Méot, *Raytracing in 3-D field maps with Zgoubi*, report DSM/LNS/GT/90-01, CEA Saclay, 1990.

[9] G. Leleux, *Compléments sur la physique des accélérateurs*, cours du DEA Grands-Instruments, Univ. Paris-VI, report IRF/LNS/86-101, CEA Saclay, March 1986.

[10] F. Méot, *A numerical method for combined spin tracking and raytracing of charged particles*, NIM **A313** (1992) 492, and proc. EPAC (1992) p.747.

[11] D. J. Kelliher et al., *Muon decay ring study*, Procs. EPAC08 Conf., Genoa, Italy (2008).

[12] F. Méot, N. Monseu, *Lattice Design and Study Tools Regarding the Super-B Project,* Procs. IPAC10 Conf., Kyoto, Japan (2010).

[13] F. Méot, *Spin tracking simulations in AGS based on ray-tracing methods,* Tech. Note C-AD/AP/452, BNL C-AD (2009) ;
F. Méot, *Zgoubi-ing AGS : spin motion with snakes and jump-quads,* Tech. Note C-AD/AP/453, BNL C-AD (2009).

[14] F. Méot, M. Bai, V. Ptitsyn, V. Ranjbar, *Spin Code Benchmarking at RHIC,* Procs. PAC11 Conf., New York (2011).

[15] F. Méot, *Raytracing Based Spin Tracking,* EDM Searches at Storage Rings Workshop, ECT - Center for Studies in Nuclear Physics and related Areas, Trento, Italy (Oct. 1-5, 2012).

[16] V. Bargmann, L. Michel, V.L. Telegdi, *Precession of the polarization of particles moving in a homogeneous electromagnetic field*, Phys. Rev. Lett. 2 (1959) 435.

[17] S.Y. Lee, Spin dynamics and snakes in synchrotrons, World Scientific, 1997.

[18] S.Y. Lee, private communication, April 2017.

[19] F. Méot and J. Payet, *Numerical tools for the simulation of synchrotron radiation loss and induced dynamical effects in high energy transport lines*, Report DSM/DAPNIA/SEA-00-01, CEA Saclay (2000).

[20] G. Leleux et al., *SR perturbations in long transport lines,* IEEE 1991 Part Acc. Conf., San Francisco (May 1991).

[21] P. Lapostolle, F. Méot, S. Valero, *A new dynamics code DYNAC for electrons, protons and heavy ions in LINACS with long accelerating elements,* 1990 LINAC Conf., Albuquerque, NM, USA.

[22] *Electron Lab for Europe,* Blue Book, CNRS-IN2P3 (1994).

[23] F. Méot, *Benchmarking stepwise ray-tracing in rings in presence of radiation damping,* Procs. PAC11 Conf., New York (2011).

[24] V. O. Kostroun, *Simple numerical evaluation of modified Bessel functions and integrals [...]*, NIM 172 (1980) 371-374.

[25] F. Méot, *Synchrotron radiation interferences at the LEP miniwiggler*, Report CERN SL/94-22 (AP), 1994.

[26] L. Ponce, R. Jung, F. Méot, *LHC proton beam diagnostics using synchrotron radiation*, Yellow Report CERN-2004-007.

[27] F. Méot, *A theory of low frequency far-field synchrotron radiation*, Particle Accelerators Vol 62, pp. 215-239 (1999).

[28] Albert Hofmann, *The Physics of Synchrotron Radiation*, Cambridge University Press, May 13, 2004.

[29] Malek Haj Tahar, *Fixed Field Ring Accelerators and ADS-Reactor Application*, PhD dissertation, Brookhaven National Laboratory and Grenoble-Alpes University (2016).

[30] B. Mayer, personal communication, CEA Saclay, Laboratoire National SATURNE, 1990.

[31] L. Farvacque *et al., Beta user's guide*, Note ESRF-COMP-87-01, 1987 ;
J. Payet, IRF/LNS, CEA Saclay, private communication ;
J.M. Lagniel, *Recherche d'un optimum*, Note IRF/LNS/SM 87/48, CEA Saclay 1987.

[32] Installed by J. S. Berg, BNL (2007). *Cf. Detection and remediation of stagnation in the Nelder-Mead algorithm using a sufficient decrease condition*, C. T. Kelley, Siam J. Optim., Vol. 10, No. 1, pp. 43-55.

[33] F. Desforges, *Implementation of a coupled treatment of the one-turn mapping in the ray-tracing code **zgoubi**,* C-AD Note

[34] F. Méot and N. Willis, *Raytrace computation with Monte Carlo simulation of particle decay*, internal report CEA/LNS/88-18 CEA Saclay, 1988.

[35] E.J. Bleser, *The parameters of the bare AGS,* Tech. Note AGS/AD?Tech. Note 430, March 15, 1996.

[36] These transfer functions have been copied from the MADX model of the AGS.

[37] H.A. Enge, *Deflecting magnets*, in **Focusing of Charged Particles**, ed. A. Septier, **Vol. II**, pp 203-264, Academic Press Inc., 1967.

[38] P. Birien et S. Valéro, *Projet de spectromètre magnétique à haute résolution pour ions lourds*, **Section IV** p.62, Note CEA-N-2215, CEA Saclay, mai 1981.

[39] Files developed by Simon White, January 2012. Including beam-beam spin kick after Ref. [40].

[40] Y. K. Batygin, *Spin depolarization due to beam-beam collisions*, Phys. Rev. E, Vol. 58, 1, July 1998.

[41] A. Hofmann, S. Myers, *Beam Dynamics in a Double RF System*, CERN ISR-TH-RF/80-26 (July 1980).

[42] F. Lemuet, F. Méot, *Developements in the ray-tracing code Zgoubi for 6-D multiturn tracking in FFAG rings*, NIM A **547** (2005) 638-651.

[43] F. Méot, *6-D beam dynamics simulations in FFAGs using the ray-tracing code Zgoubi*, ICFA Beam Dyn.Newslett.43:44-50 (2007).

[44] Franck Lemuet, *Collection and muon acceleration in the neutrino factory project*, PhD thesis, Paris KI University, April 2007.

[45] V. M. Kel'man and S. Ya. Yavor, *Achromatic quadrupole electron lenses,* Soviet Physics - Technical Physics, vol. 6, No 12, June 1962 ;
S. Ya. Yavor *et als., Achromatic quadrupole lenses*, NIM **26** (1964) 13-17.

[46] A. Septier, *Cours du DEA de physique des particules, optique corpusculaire,* Université d'Orsay, 1966-67, pp. 38-39.

[47] S. P. Karetskaya et als., *Mirror-bank energy analyzers, in Advances in electronics and electron physics*, Vol. 89, Acad. Press (1994) 391-491.

[48] A. Septier et J. van Acker, *Les lentilles quadrupolaires électriques*, NIM **13** (1961) 335-355 ;
Y. Fujita and H. Matsuda, *Third order transfer matrices for an electrostatic quadrupole lens,* NIM **123** (1975) 495-504.

[49] J. Fourrier, F. Martinache, F. Méot, J. Pasternak, *Spiral FFAG lattice design tools, application to 6-D tracking in a proton-therapy class lattice*, NIM A 589 (2008).

[50] F. Méot, *Tracking studies regarding EMMA FFAG project,* Internal report CEA DAPNIA-06-04 (2006).

[51] J. S. Berg et al., *Recent developments on the EMMA on-line commissioning software,* Procs. IPAC10 Conf., Kyoto, Japan (2010).

[52] S. Machida et al., *Acceleration in the linear non-scaling fixed-field alternating-gradient accelerator EMMA,* Nature Physics, vol. 8, March 2012.

[53] Installed by Pavel Akishin, JINR, Dubna, 1992.

[54] M.W. Garrett, *Calculation of fields [...] by elliptic integrals*, J. Appl. Phys., **34**, 9, sept. 1963.

[55] P.F. Byrd and M.D. Friedman, *Handbook of elliptic integrals for engineers and scientists,* pp. 282-283, Springer-Verlag, Berlin, 1954.

[56] Installation by M. Bai, BNL, 2009.

[57] A. Tkatchenko, *Computer program UNIPOT*, SATURNE, CEA Saclay, 1982.

[58] S.B. Kowalski, H.A. Enge, The ion-optical program raytrace, NIM A258 Vol. 3 (1987) 407.

[59] N. Tsoupas, private communication, Brookhaven National Laboratory, Oct 2015.

[60] J.L. Chuma, *PLOTDATA*, TRIUMF Design Note TRI-CO-87-03a.

[61] F. Méot et la., A model of the AGS in the ray-tracing code Zgoubi, Tech. Note C-A/AP/***, 2013.

[62] J. Thirion et P. Birien, *Le spectromètre II, Internal Report DPh-N/ME*, CEA Saclay, 23 Déc. 1975 ;
H. Catz, *Le spectromètre SPES II*, Internal Report DPh-N/ME, CEA Saclay, 1980 ;
A. Moalem, F. Méot, G. Leleux, J.P. Penicaud, A. Tkatchenko, P. Birien, *A modified QDD spectrometer for η meson decay measurements,* NIM A289 (1990) 168-175

[63] P. Pile, I-H. Chiang, K. K. Li, C. J. Kost, J. Doornbos, F. Méot et als., *A two-stage separated 800-MeV/c Kaon beamline*, TRIUMF and BNL Preprint (1997).

[64] F. Méot, *The raytracing code Zgoubi*, CERN SL/94-82 (AP) (1994), 3rd Intern. Workshop on Optimization and Inverse Problems in Electromagnetism, CERN, Geneva, Switzerland, 19-21 Sept. 1994.

[65] E. Grorud, J.L. Laclare, G. Leleux, *Résonances de dépolarisation dans SATURNE 2*, Int. report GOC-GERMA 75-48/TP-28, CEA Saclay (1975), and, Home Computer Codes POLAR and POPOL, IRF/LNS/GT, CEA Saclay (1975).

[66] M. Froissart et R. Stora, *Dépolarisation d'un faisceau de protons polarisés dans un synchrotron*, NIM 7 (1960) 297-305.

[67] J.L. Hamel, *mini graphic library LIBMINIGRAF*, CEA-DSM, Saclay, 1996.

[68] F. Méot e , A. París, *Concerning effects of fringe fields and longitudinal distribution of b10 in low-β regions on dynamics in LHC,* report FERMILAB-TM-2017, August 23, 1997.

[69] A white paper: The Cornell-BNL FFAG-ERL Test Accelerator, I. Ben-Zvi et al., Dec. 16, 2014.

[70] F. Méot, *End-to-end 9-D polarized bunch transport in FFAG eRHIC*, EIC14 workshop, http://www.jlab.org/conferences/eic2014/ (2014).

[71] http://www.gnuplot.info/

Index