Fermi National Accelerator Laboratory

FERMILAB-TM-2010
DAPNIA/SEA-97-13

Zgoubi Users' Guide
Version 4

Francios Méot

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

S. Valéro

CEA DSM/LNS/GECA, CE-Saclay
91191 Gif-sur-Yvette10/27/97, France

October 1997

Operated by Universities Research Association Inc. under Contract No. DE-AC02-76CH03000 with the United States Department of Energy

DAPNIA/SEA-97-13
FERMILAB-TM-2010

ZGOUBI USERS' GUIDE

– VERSION 4 –

F. Méot *

Fermi National Accelerator Laboratory, BD/Physics

P.O. Box 500, Batavia, IL 60510, USA

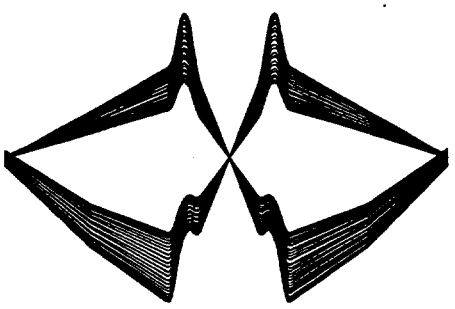
and

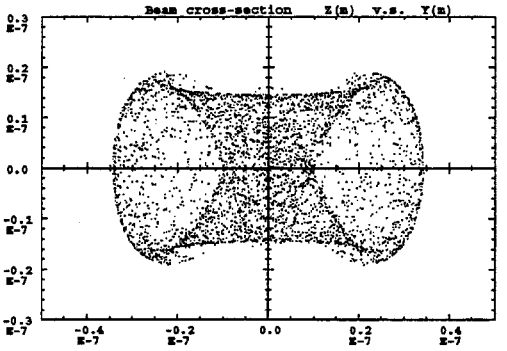
S. Valéro

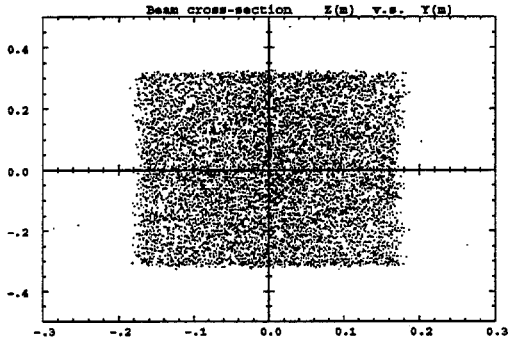
CEA, DSM/LNS/GECA, CE-SACLAY

91191 Gif-sur-Yvette, France

October 15, 1997







*Present address : CEA, DSM/DAPNIA/SEA, CE-SACLAY, 91191 Gif-sur-Yvette, France

Table of contents

PART A Description of software contents 1

GLOSSARY OF KEYWORDS 3

OPTICAL ELEMENTS VERSUS KEYWORDS 5

1 INTRODUCTION 7

2 NUMERICAL CALCULATION OF MOTION AND FIELDS 9
- 2.1 Zgoubi Frame . . . 9
- 2.2 Integration of Lorentz Equation . . . 9
  - 2.2.1 Integration in magnetic fields . . . 10
  - 2.2.2 Integration in electric fields . . . 11
  - 2.2.3 Integration in combined electric and magnetic fields . . . 13
- 2.3 Calculation of $\vec{B}$ and its Derivatives . . . 14
  - 2.3.1 Extrapolation from a 1-D axial field map . . . 14
  - 2.3.2 Extrapolation from Median Plane Fields . . . 14
  - 2.3.3 Extrapolation from arbitrary 2-D Field Maps . . . 15
  - 2.3.4 Interpolation in 3-D Field Maps . . . 15
  - 2.3.5 3-D Analytical Models of Fields . . . 15
- 2.4 Calculation of $\vec{B}$ from Field Maps . . . 16
  - 2.4.1 1-D Axial Map, with Cylindrical Symmetry . . . 16
  - 2.4.2 2-D Median Plane Map, with Median Plane Antisymmetry . . . 16
  - 2.4.3 Arbitrary 2-D Map, no Symmetries . . . 19
  - 2.4.4 Calculation of $\vec{B}$ from a 3-D Field Map . . . 20
- 2.5 Calculation of $\vec{E}$ and its derivatives . . . 21
  - 2.5.1 Extrapolation from a 1-D axial field map . . . 21
  - 2.5.2 Extrapolation from analytically defined axial fields . . . 22
  - 2.5.3 3-D Analytical models of fields . . . 22
- 2.6 Calculation of $\vec{E}$ from field maps . . . 22

3 SPIN TRACKING 22

4 SYNCHROTRON RADIATION [10] 24
- 4.1 Calculation of the electric field $\mathcal{E}(\vec{n}, \tau)$ . . . 24
- 4.2 Calculation of the Fourier transform of the electric field . . . 26

5 DESCRIPTION OF THE AVAILABLE PROCEDURES 27
- 5.1 Introduction . . . 27
- 5.2 Definition of an Object . . . 27
- 5.3 Declaration of options . . . 35
- 5.4 Optical Elements and related numerical procedures . . . 55
- 5.5 Output Procedures . . . 100
- 5.6 Complementary Features . . . 108
  - 5.6.1 Backward Ray-tracing . . . 108
  - 5.6.2 Checking Fields and Trajectories in Magnets . . . 108
  - 5.6.3 Labeling keywords . . . 108
  - 5.6.4 Multiturn tracking in circular machines . . . 109
  - 5.6.5 Positioning of Magnets and field maps . . . 109
  - 5.6.6 Coded integration step . . . 109
  - 5.6.7 Ray-tracing of an arbitrarily large number of particles . . . 111
  - 5.6.8 Stopped particles: the IEX flag . . . 111

5.6.9 Negative rigidity 111

PART B Keywords and input data formatting 113

GLOSSARY OF KEYWORDS 115

OPTICAL ELEMENTS VERSUS KEYWORDS 117

INTRODUCTION 119

PART C Examples of input data files and output result files 191

INTRODUCTION 193

1 MONTE CARLO IMAGES IN SPES 2 195

2 TRANSFER MATRICES ALONG A TWO STAGE SEPARATION KAON BEAM LINE 198

3 IN-FLIGHT DECAY IN SPES 3 201

5 USE OF THE FITTING PROCEDURE 203

6 MULTITURN SPIN TRACKING IN SATURNE 205

7 MICRO-BEAM FOCUSING WITH ELECTROMAGNETIC QUADRUPOLES 207

PART D Running zgoubi and its post-processor/graphic interface zgplot 211

1 INTRODUCTION 213

2 GETTING TO RUN zgoubi AND zgplot 213
2.1 Making the executable files zgoubi and zgplot 213
2.1.1 The transportable package zgoubi 213
2.1.2 The post-processor and graphic interface package zgplot 213
2.2 Running zgoubi 213
2.3 Running zgplot 213

3 STORAGE FILES 213

REFERENCES 216

INDEX 217

PART A

Description of software contents

Glossary of keywords

AIMANT	Generation of a dipole magnet 2-D map	55
AUTOREF	Automatic transformation to a new reference frame	60
BEND	Bending magnet	61
BINARY	BINARY/FORMATTED data converter	36
BREVOL	1-D uniform mesh magnetic field map	62
CARTEMES	2-D Cartesian uniform mesh magnetic field map	63
CAVITE	Accelerating cavity	65
CHAMBR	Long transverse aperture limitation	67
CHANGREF	Transformation to a new reference frame	68
CIBLE	Generate a secondary beam from target interaction	69
CLORB	Beam centroid path; closed orbit	101
COLLIMA	Collimator	70
DECAPOLE	Decapole magnet	71
DIPOLE	Generation of a dipole magnet 2-D map	72
DODECAPO	Dodecapole magnet	74
DRIFT	Field free drift space	75
EBMULT	Electro-magnetic multipole	76
EL2TUB	Two-tubes electrostatic lens	77
ELMULT	Electric multipole	78
ELREVOL	1-D uniform mesh electric field map	80
END	End of input data list ; see FIN	37
ESL	Field free drift space	75
FAISCEAU	Print particle coordinates	102
FAISCNL	Store particle coordinates in file FNAME	102
FAISCNLA	Store coordinates every IP other pass at labeled elements	102
FIN	End of input data list	37
FIT	Fitting procedure	38
FOCALE	Particle coordinates and horizontal beam dimension at distance XL	103
FOCALEZ	Particle coordinates and vertical beam dimension at distance XL	103
GASCAT	Gas scattering	42
HISTO	1-D histogram	104
IMAGE	Localization and size of horizontal waist	103
IMAGES	Localization and size of horizontal waists	103
IMAGESZ	Localization and size of vertical waists	103
IMAGEZ	Localization and size of vertical waist	103
MAP2D	2-D Cartesian uniform mesh magnetic field map without symmetry	81
MATRIX	Calculation of transfer coefficients	105
MCDESINT	Monte-Carlo simulation of in-flight decay	43
MCOBJET	Monte-Carlo generation of a 3-D object	28
MULTIPOL	Magnetic multipole	82
OBJET	Generation of an object	31
OBJETA	Object from Monte-Carlo simulation of decay reaction	34
OCTUPOLE	Octupole magnet	83
ORDRE	Higher order Taylor expansions in lens	46
PARTICUL	Particle characteristics	47
PLOTDATA	Intermediate outputs for the PLOTDATA computer graphic software	106
POISSON	Read field data from *POISSON* output	84
POLARMES	2-D polar mesh field map	85
PS170	Simulation of a round shape dipole magnet	86
QUADISEX	Sharp edge magnetic multipoles	87
QUADRUPO	Quadrupole magnet	88

REBELOTE Jump to the beginning of Zgoubi input data file 48
RESET Reset counters and flags 49
SCALING Time scaling of power supplies and R.F. 50
SEPARA Wien Filter - analytic simulation 90
SEXQUAD Sharp edge magnetic multipole 87
SEXTUPOL Sextupole magnet 91
SOLENOID Solenoid 92
SPNPRNL Store spin coordinates into file FNAME 107
SPNPRNLA Store spin coordinates every *IP* other pass 107
SPNPRT Print spin coordinates 107
SPNTRK Spin tracking 52
SYNRAD Synchrotron radiation 54
TARGET Generate a secondary beam from target interaction ; see CIBLE 69
TOSCA 2-D and 3-D Cartesian uniform mesh field map 93
TRANSMAT Matrix transfer 94
TRAROT Translation-Rotation of the reference frame 95
UNIPOT Unipotential electrostatic lens 96
VENUS Simulation of a rectangular dipole magnet 97
WIENFILT Wien filter 98
YMY Reverse signs of Y and Z axes 99

Optical elements versus keywords

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

Field map procedures are also cataloged; in most cases an adequate field map can be used for simulating these elements.

MAGNETIC ELEMENTS

Decapole	DECAPOLE, MULTIPOL
Dipole	AIMANT, BEND, DIPOLE, MULTIPOL, QUADISEX
Dodecapole	DODECAPO, MULTIPOL
Multipole	MULTIPOL, QUADISEX, SEXQUAD
Octupole	OCTUPOLE, MULTIPOL, QUADISEX, SEXQUAD
Quadrupole	QUADRUPO, MULTIPOL, SEXQUAD
Sextupole	SEXTUPOL, MULTIPOL, QUADISEX, SEXQUAD
Skewed multipoles	MULTIPOL
Solenoid	SOLENOID

Field maps

1-D, cylindrical symmetry	BREVOL
2-D, mid-plane symmetry	CARTEMES, POISSON, TOSCA
2-D, no symmetry	MAP2D
3-D	TOSCA

ELECTRIC ELEMENTS

Decapole	ELMULT
Dipole	ELMULT
Dodecapole	ELMULT
Multipole	ELMULT
Octupole	ELMULT
Quadrupole	ELMULT
R.F. cavity	CAVITE
Sextupole	ELMULT
Skewed multipoles	ELMULT
2-tube (bipotential) lens	EL2TUB
3-tube (unipotential) lens	UNIPOT

Field maps

1D, cylindrical symmetry	ELREVOL

ELECTROMAGNETIC ELEMENTS

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

1 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 and other magnetic or electric devices, and is able as well to handle periodic structures. Compared to other codes, it presents several peculiarities:

- 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,
- 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) field maps,
- Monte Carlo procedures: unlimited number of trajectories, in-flight decay, etc.
- a built-in fitting procedure,
- multiturn tracking in circular accelerators including many features proper to machine parameter calculation and survey, and also the simulation of time-varying power supplies.

The initial version of the Code, dedicated to the ray-tracing in magnetic fields, was developed by D. Garreta and J.C. Faivre at CEN-Saclay in the early 1970's. 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 now in use in several national and foreign laboratories.

The first manual was in French [1]. Since then many improvements have been implemented. 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 was necessary for accompanying the third version of the code which featured spin tracking and ray-tracing in combined electric and magnetic fields; this was done with the help of D. Bunel for the preparation of the document and lead to the third release [3].

Lately, provisions were introduced for the computation of synchrotron radiation electromagnetic impulse and spectra. 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 benefited from extensive development of storage ring related features.

The graphic interface to **Zgoubi** (Part D) has also undergone concomitent extended developments, which make it a performant tool for post-processing **Zgoubi** outputs.

These recent developments of **Zgoubi** [1, 2, 3, and the present version of the guide] have strongly benefited of the environment of the Groupe Théorie, Laboratoire National SATURNE, CEA/DSM-Saclay.

This manual is intended only to describe the details of the most recent version of **Zgoubi**, which is far from being a "finished product".

2 NUMERICAL CALCULATION OF MOTION AND FIELDS

2.1 Zgoubi Frame

The reference frame of **Zgoubi** is presented in Fig 1. Its origin is in the median plane on a reference curve which coincides with the optical axis of optical elements.

2.2 Integration of Lorentz Equation

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

$$\frac{d(m\vec{v})}{dt} = q\,(\vec{e} + \vec{v} \times \vec{b}) \tag{2.2.1}$$

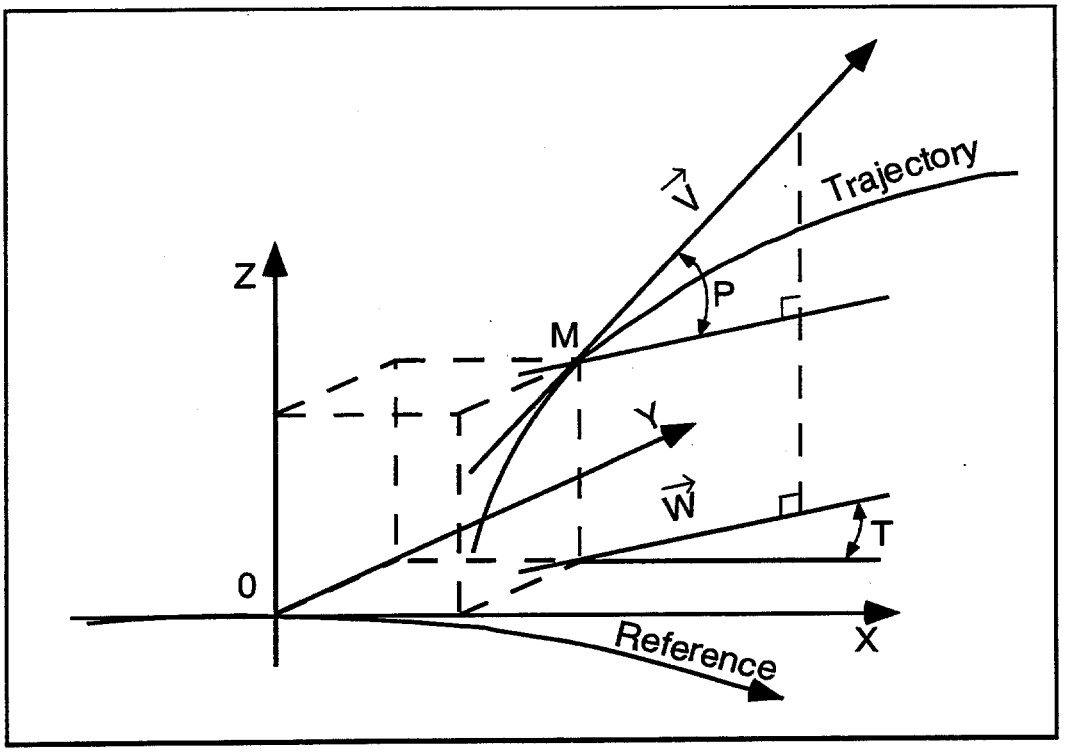


Figure 1: Reference frame and coordinates (Y, T, Z, P) in **Zgoubi**.
OX: in the plane of the reference curve in the direction of motion,
OY: in the plane of the reference curve, normal to OX,
OZ: orthogonal to the (X, Y) plane,
$\vec{W}$: projection of the velocity, $\vec{v}$, in the (X, Y) plane,
T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

Taking

$$\vec{u} = \frac{\vec{v}}{v}, \quad ds = v\,dt, \quad \vec{u}' = \frac{d\vec{u}}{ds}, \quad m\vec{v} = mv\vec{u} = qB\rho\vec{u} \tag{2.2.2}$$

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

$$(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} + \vec{u} \times \vec{b} \tag{2.2.3}$$

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 given by Taylor expansions (Fig. 2)

$$\begin{aligned} \vec{R}(M_1) &= \vec{R}(M_0) + \vec{u}(M_0)\,\Delta s + \vec{u}'(M_0)\,\frac{\Delta s^2}{2!} + \ldots + \vec{u}''''(M_0)\,\frac{\Delta s^6}{6!} \\ \vec{u}(M_1) &= \vec{u}(M_0) + \vec{u}'(M_0)\,\Delta s + \vec{u}''(M_0)\,\frac{\Delta s^2}{2!} + \ldots + \vec{u}''''(M_0)\,\frac{\Delta s^5}{5!} \end{aligned} \tag{2.2.4}$$

The rigidity at M_1 is obtained in the same way from

$$(B\rho)(M_1) = (B\rho)(M_0) + (B\rho)'(M_0)\Delta s + \ldots + (B\rho)''''(M_0)\frac{\Delta s^4}{4!} \tag{2.2.5}$$

The derivatives $\vec{u}^{(n)} = \frac{d^n\vec{u}}{ds^n}$ and $(B\rho)^{(n)} = \frac{d^n(B\rho)}{ds^n}$ 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.

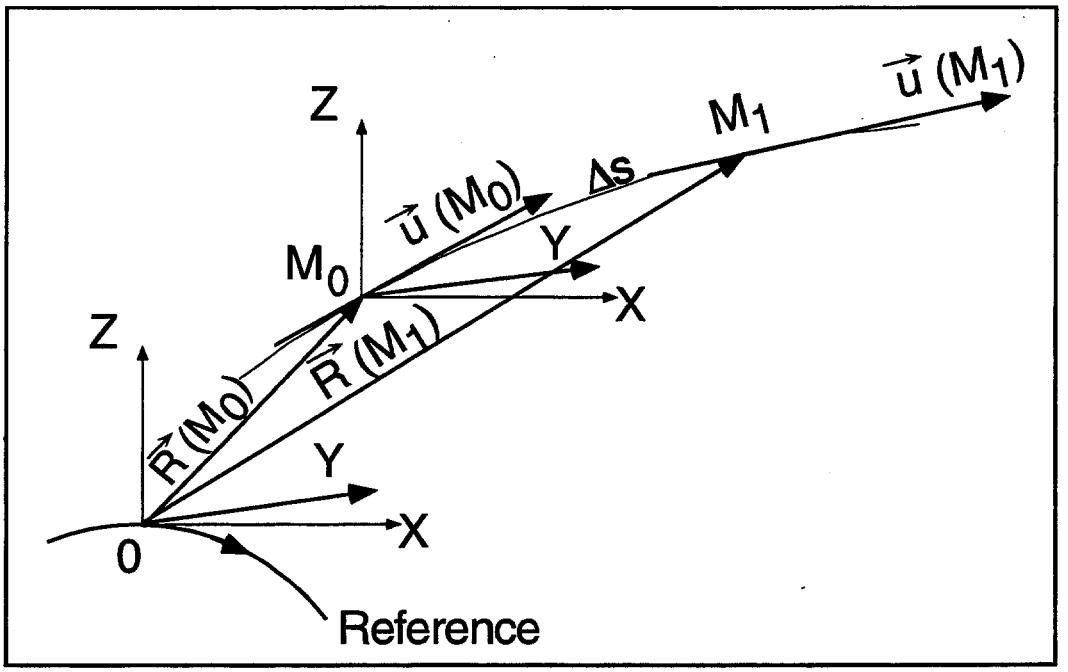


Figure 2: Position and velocity of a particle in the reference frame.

2.2.1 Integration in magnetic fields

Admitting that $\vec{e} = 0$, and noting $\vec{B} = \frac{\vec{b}}{B\rho}$, eq. (2.2.3) reduces to

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

The successive derivatives $\vec{u}^{(n)} = \frac{d^n\vec{u}}{ds^n}$ of $\vec{u}$ needed in the Taylor expansions (Eqs. 2.2.4) are calculated by differentiating $\vec{u}' = \vec{u} \times \vec{B}$

$$
\begin{aligned}
\vec{u}'' &= \vec{u}' \times \vec{B} + \vec{u} \times \vec{B}' \\
\vec{u}''' &= \vec{u}'' \times \vec{B} + 2\vec{u}' \times \vec{B}' + \vec{u} \times \vec{B}'' \\
\vec{u}'''' &= \vec{u}''' \times \vec{B} + 3\vec{u}'' \times \vec{B}' + 3\vec{u}' \times \vec{B}'' + \vec{u} \times \vec{B}''' \\
\vec{u}''''' &= \vec{u}'''' \times \vec{B} + 4\vec{u}''' \times \vec{B}' + 6\vec{u}'' \times \vec{B}'' + 4\vec{u}' \times \vec{B}''' + \vec{u} \times \vec{B}''''
\end{aligned}
\tag{2.2.6}
$$

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

From $d\vec{B} = \dfrac{\partial \vec{B}}{\partial X}\, dX + \dfrac{\partial \vec{B}}{\partial Y}\, dY + \dfrac{\partial \vec{B}}{\partial Z}\, dZ = \displaystyle\sum_{i=1,3} \frac{\partial \vec{B}}{\partial X_i}\, dX_i$, and by successive differentiation, we get

$$
\begin{aligned}
\vec{B}' &= \sum_i \frac{\partial \vec{B}}{\partial X_i}\, u_i \\
\vec{B}'' &= \sum_{ij} \frac{\partial^2 \vec{B}}{\partial X_i \partial X_j}\, u_i u_j + \sum_i \frac{\partial \vec{B}}{\partial X_i}\, u_i' \\
\vec{B}''' &= \sum_{ijk} \frac{\partial^3 \vec{B}}{\partial X_i \partial X_j \partial X_k}\, u_i u_j u_k + 3 \sum_{ij} \frac{\partial^2 \vec{B}}{\partial X_i \partial X_j}\, u_i' u_j + \sum_i \frac{\partial \vec{B}}{\partial X_i}\, u_i'' \\
\vec{B}'''' &= \sum_{ijkl} \frac{\partial^4 \vec{B}}{\partial X_i \partial X_j \partial X_k \partial X_l}\, u_i u_j u_k u_l + 6 \sum_{ijk} \frac{\partial^3 \vec{B}}{\partial X_i \partial X_j \partial X_k}\, u_i' u_j u_k \\
&\quad + 4 \sum_{ij} \frac{\partial^2 \vec{B}}{\partial X_i \partial X_j}\, u_i'' u_j + 3 \sum_{ij} \frac{\partial^2 \vec{B}}{\partial X_i \partial X_j}\, u_i' u_j' + \sum_i \frac{\partial \vec{B}}{\partial X_i}\, u_i'''
\end{aligned}
\tag{2.2.7}
$$

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. (2.2.6) and (2.2.7), and inject it in Eq. (2.2.4) to get $\vec{R}(M_1)$ and $\vec{u}(M_1)$.

2.2.2 Integration in electric fields [4]

Admitting that $\vec{b} = 0$, eq. (2.2.3) reduces to

$$
(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} \tag{2.2.8}
$$

which, by successive differentiations, gives the recursive relations

$$
\begin{aligned}
&(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} \\
&(B\rho)''\vec{u} + 2(B\rho)'\vec{u}' + B\rho\vec{u}'' = \left(\frac{1}{v}\right)' \vec{e} + \frac{\vec{e}'}{v} \\
&(B\rho)'''\vec{u} + 3(B\rho)''\vec{u}' + 3(B\rho)'\vec{u}'' + B\rho\vec{u}''' = \left(\frac{1}{v}\right)'' \vec{e} + 2\left(\frac{1}{v}\right)' \vec{e}' + \left(\frac{1}{v}\right) \vec{e}'' \\
&(B\rho)''''\vec{u} + 4(B\rho)'''\vec{u}' + 6(B\rho)''\vec{u}'' + 4(B\rho)'\vec{u}''' + B\rho\vec{u}'''' = \\
&\qquad \left(\frac{1}{v}\right)''' \vec{e} + 3\left(\frac{1}{v}\right)'' \vec{e}' + 3\left(\frac{1}{v}\right)' \vec{e}'' + \frac{1}{v}\vec{e}'''
\end{aligned}
\tag{2.2.9}
$$

that provide the derivatives $\dfrac{d^n \vec{u}}{ds^n}$ needed in the Taylor expansions (2.2.4)

$$
\begin{aligned}
\vec{u}' &= \left(\frac{1}{v}\right)\vec{E} - \frac{(B\rho)'}{B\rho}\vec{u} \\
\vec{u}'' &= \left(\frac{1}{v}\right)'\vec{E} + \left(\frac{1}{v}\right)\vec{E}'\,|_{B\rho} - 2\frac{(B\rho)'}{B\rho}\vec{u}' - \frac{(B\rho)''}{B\rho}\vec{u} \\
\vec{u}''' &= \left(\frac{1}{v}\right)''\vec{E} + 2\left(\frac{1}{v}\right)'\vec{E}'\,|_{B\rho} + \frac{1}{v}\vec{E}''\,|_{B\rho} - 3\frac{(B\rho)'}{B\rho}\vec{u}'' - 3\frac{(B\rho)''}{B\rho}\vec{u}' - \frac{(B\rho)'''}{B\rho}\vec{u} \\
\vec{u}'''' &= \left(\frac{1}{v}\right)'''\vec{E} + 3\left(\frac{1}{v}\right)''\vec{E}'\,|_{B\rho} + 3\left(\frac{1}{v}\right)'\vec{E}''\,|_{B\rho} + \left(\frac{1}{v}\right)\vec{E}'''\,|_{B\rho} \\
&\quad - 4\frac{(B\rho)'}{B\rho}\vec{u}''' - 6\frac{(B\rho)''}{B\rho}\vec{u}'' - 4\frac{(B\rho)'''}{B\rho}\vec{u}' - \frac{(B\rho)''''}{B\rho}\vec{u}
\end{aligned}
\tag{2.2.10}
$$

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

$$
d\vec{E} = \frac{\partial\vec{E}}{\partial X}\,dX + \frac{\partial\vec{E}}{\partial Y}\,dY + \frac{\partial\vec{E}}{\partial Z}\,dZ \tag{2.2.11}
$$

by successive differentiations

$$
\begin{aligned}
\vec{E}' &= \sum_i \frac{\partial\vec{E}}{\partial X_i}\,u_i \\
\vec{E}'' &= \sum_{ij} \frac{\partial^2\vec{E}}{\partial X_i \partial X_j}\,u_i u_j + \sum_i \frac{\partial\vec{E}}{\partial X_i}\,u_i' \\
\vec{E}''' &= \sum_{ijk} \frac{\partial^3\vec{E}}{\partial X_i \partial X_j \partial X_k}\,u_i u_j u_k + 3\sum_{ij} \frac{\partial^2\vec{E}}{\partial X_i \partial X_j}\,u_i' u_j + \sum_i \frac{\partial\vec{E}}{\partial X_i}\,u_i''
\end{aligned}
\tag{2.2.12}
$$

These eq. (2.2.10), as well as the calculation of the rigidity, following eq. (2.2.5), involve derivatives $(B\rho)^{(n)} = \dfrac{d^n(B\rho)}{ds^n}$, which are obtained in the following way. Considering that

$$
\frac{dp^2}{dt} = \frac{d\vec{p}^2}{dt} \quad i.e., \quad \frac{dp}{dt}\,p = \frac{d\vec{p}}{dt}\vec{p} \tag{2.2.13}
$$

with $\dfrac{d\vec{p}}{dt} = q\,(\vec{e} + \vec{v}\times\vec{b})$ (eq. 2.2.1), we obtain

$$
\frac{dp}{dt}\,p = q\,(\vec{e} + v\times\vec{b})\cdot\vec{p} = q\vec{e}\cdot\vec{p} \tag{2.2.14}
$$

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. (2.2.14) leads to the $(B\rho)^{(n)}$

$$
\begin{aligned}
(B\rho)' &= \frac{1}{v}(\vec{e}\cdot\vec{u}) \\
(B\rho)'' &= \left(\frac{1}{v}\right)'(\vec{e}\cdot\vec{u}) + \frac{1}{v}(\vec{e}\cdot\vec{u})' \\
(B\rho)''' &= \left(\frac{1}{v}\right)''(\vec{e}\cdot\vec{u}) + 2\left(\frac{1}{v}\right)'(\vec{e}\cdot\vec{u})' + \frac{1}{v}(\vec{e}\cdot\vec{u})'' \\
(B\rho)'''' &= \left(\frac{1}{v}\right)'''(\vec{e}\cdot\vec{u}) + 3\left(\frac{1}{v}\right)''(\vec{e}\cdot\vec{u})' + 3\left(\frac{1}{v}\right)'(\vec{e}\cdot\vec{u})'' + \frac{1}{v}(\vec{e}\cdot\vec{u})'''
\end{aligned}
\tag{2.2.15}
$$

Note that the derivatives $(\vec{e}\cdot\vec{u})^{(n)} = \dfrac{d^n(\vec{e}\cdot\vec{u})}{ds^n}$ 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

$$
\frac{d^{n+1}W}{ds^{n+1}} = q\frac{d^n(\vec{e}\cdot\vec{u})}{ds^n}
\tag{2.2.16}
$$

Finally, the derivatives $\left(\dfrac{1}{v}\right)^{(n)} = \dfrac{d^n\left(\dfrac{1}{v}\right)}{ds^n}$ involved in eqs (2.2.10,2.2.15) are obtained from $p = \dfrac{v}{c}\dfrac{W+mc^2}{c}$, by successive differentiations, that give the recursive relations

$$
\begin{aligned}
\left(\frac{1}{v}\right) &= \frac{1}{c^2}\frac{W+mc^2}{qB\rho} \\
\left(\frac{1}{v}\right)' &= \frac{1}{c^2}\frac{(\vec{e}\cdot\vec{u})}{B\rho} - \frac{1}{v}\frac{(B\rho)'}{B\rho} \\
\left(\frac{1}{v}\right)'' &= \frac{1}{c^2}\frac{(\vec{e}\cdot\vec{u})'}{B\rho} - 2\left(\frac{1}{v}\right)'\frac{(B\rho)'}{B\rho} - \frac{1}{v}\frac{(B\rho)''}{B\rho} \\
\left(\frac{1}{v}\right)''' &= \frac{1}{c^2}\frac{(\vec{e}\cdot\vec{u})''}{B\rho} - 3\left(\frac{1}{v}\right)''\frac{(B\rho)'}{B\rho} - 3\left(\frac{1}{v}\right)'\frac{(B\rho)''}{B\rho} - \frac{1}{v}\frac{(B\rho)'''}{B\rho}
\end{aligned}
\tag{2.2.17}
$$

2.2.3 Integration in combined electric and magnetic fields

When both $\vec{e}$ and $\vec{b}$ are non-zero, the complete eq. (2.2.3) must be considered. Successive differentiations give the following recursive relations

$$
\begin{aligned}
&(B\rho)'\vec{u} + B\rho\vec{u}' = \frac{\vec{e}}{v} + \vec{u}\times\vec{b} \\
&(B\rho)''\vec{u} + 2(B\rho)'\vec{u}' + B\rho\vec{u}'' = \left(\frac{1}{v}\right)'\vec{e} + \left(\frac{1}{v}\right)\vec{e}' + (\vec{u}\times\vec{b})' \\
&(B\rho)'''\vec{u} + 3(B\rho)''\vec{u}' + 3(B\rho)'\vec{u}'' + B\rho\vec{u}''' = \left(\frac{1}{v}\right)''\vec{e} + 2\left(\frac{1}{v}\right)'\vec{e}' + \left(\frac{1}{v}\right)\vec{e}'' + (\vec{u}\times\vec{b})'' \\
&(B\rho)''''\vec{u} + 4(B\rho)'''\vec{u}' + 6(B\rho)''\vec{u}'' + 4(B\rho)'\vec{u}''' + B\rho\vec{u}'''' = \\
&\qquad \left(\frac{1}{v}\right)'''\vec{e} + 3\left(\frac{1}{v}\right)''\vec{e}' + 3\left(\frac{1}{v}\right)'\vec{e}'' + \frac{1}{v}\vec{e}''' + (\vec{u}\times\vec{b})'''
\end{aligned}
\tag{2.2.18}
$$

that provide the derivatives $\dfrac{d^n\vec{u}}{ds^n}$ needed in the Taylor expansions (2.2.4)

$$
\begin{aligned}
\vec{u}' &= \left(\frac{1}{v}\right)\vec{E} + (\vec{u}\times\vec{B}) - \frac{(B\rho)'}{B\rho}\vec{u} \\
\vec{u}'' &= \left(\frac{1}{v}\right)'\vec{E} + \left(\frac{1}{v}\right)\vec{E}'\,|_{B\rho} + (\vec{u}\times\vec{B}')'\,|_{B\rho} - 2\frac{(B\rho)'}{B\rho}\vec{u}' - \frac{(B\rho)''}{B\rho}\vec{u} \\
\vec{u}''' &= \left(\frac{1}{v}\right)''\vec{E} + 2\left(\frac{1}{v}\right)'\vec{E}'\,|_{B\rho} + \frac{1}{v}\vec{E}''\,|_{B\rho} + (\vec{u}\times\vec{B})''\,|_{B\rho} - 3\frac{(B\rho)'}{B\rho}\vec{u}'' - 3\frac{(B\rho)''}{B\rho}\vec{u}' - \frac{(B\rho)'''}{B\rho}\vec{u} \\
\vec{u}'''' &= \left(\frac{1}{v}\right)'''\vec{E} + 3\left(\frac{1}{v}\right)''\vec{E}'\,|_{B\rho} + 3(\frac{1}{v})'\vec{E}''\,|_{B\rho} + \left(\frac{1}{v}\right)\vec{E}'''\,|_{B\rho} \\
&\quad + (\vec{u}\times\vec{B})'''\,|_{B\rho} - 4\frac{(B\rho)'}{B\rho}\vec{u}''' - 6\frac{(B\rho)''}{B\rho}\vec{u}'' - 4\frac{(B\rho)'''}{B\rho}\vec{u}' - \frac{(B\rho)''''}{B\rho}\vec{u}
\end{aligned}
\tag{2.2.19}
$$

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

$$
\vec{E}^{(n)}\,|_{B\rho} = \frac{1}{B\rho}\frac{d^n\vec{e}}{ds^n} \quad \text{and} \quad (\vec{u}\times\vec{B})^{(n)}\,|_{B\rho} = \frac{1}{B\rho}(\vec{u}\times\vec{b})^{(n)}. \tag{2.2.20}
$$

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. (2.2.7) and (2.2.12).

2.3 Calculation of $\vec{B}$ and its Derivatives

Zgoubi calculates $\vec{B}(X,Y,Z)$ and its derivatives in several different ways, depending on whether field maps or analytic representations of optical elements are used. The five basic means are the following.

2.3.1 Extrapolation from a 1-D axial field map [4]

A cylindrically symmetric field (e.g., using *BREVOL*) 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 2.4.1). 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 to 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

$$
\begin{aligned}
B_X(X,r) &= B_X(X,0) - \frac{r^2}{4}\frac{\partial^2 B_X}{\partial X^2}(X,0) + \frac{r^4}{64}\frac{\partial^4 B_X}{\partial X^4}(X,0) \\
B_r(X,r) &= -\frac{r}{2}\frac{\partial B_X}{\partial X}(X,0) + \frac{r^3}{16}\frac{\partial^3 B_X}{\partial X^3}(X,0) - \frac{r^5}{384}\frac{\partial^5 B_X}{\partial X^5}(X,0)
\end{aligned}
\tag{2.3.1}
$$

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 eqs. (2.2.7).

2.3.2 Extrapolation from Median Plane Fields

In the median plane, $B_Z(X,Y,0)$, and its derivatives with respect to X or Y, may be calculated 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

$$
\begin{aligned}
B_X(X,Y,0) &= 0 \\
B_Y(X,Y,0) &= 0 \\
B_X(X,Y,Z) &= -B_X(X,Y,-Z) \\
B_Y(X,Y,Z) &= -B_Y(X,Y,-Z) \\
B_Z(X,Y,Z) &= B_Z(X,Y,-Z)
\end{aligned}
\tag{2.3.2}
$$

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

$$
\begin{aligned}
B_X(X,Y,Z) &= Z\frac{\partial B}{\partial X} - \frac{Z^3}{6}\left(\frac{\partial^3 B}{\partial X^3} + \frac{\partial^3 B}{\partial X \partial Y^2}\right) \\
B_Y(X,Y,Z) &= Z\frac{\partial B}{\partial Y} - \frac{Z^3}{6}\left(\frac{\partial^3 B}{\partial X^2 \partial Y} + \frac{\partial^3 B}{\partial Y^3}\right) \\
B_Z(X,Y,Z) &= B - \frac{Z^2}{2}\left(\frac{\partial^2 B}{\partial X^2} + \frac{\partial^2 B}{\partial Y^2}\right) + \frac{Z^4}{24}\left(\frac{\partial^4 B}{\partial X^4} + 2\frac{\partial^4 B}{\partial X^2 \partial Y^2} + \frac{\partial^4 B}{\partial Y^4}\right)
\end{aligned}
\tag{2.3.3}
$$

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 2.4.2) so as to get the expressions involved in eq. (2.2.7).

2.3.3 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 2.4.3 and keyword *MAP2D*).

2.3.4 Interpolation in 3-D Field Maps [5]

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 a 3-D $3 \times 3 \times$ 3-point grid (see section 2.4.4).

2.3.5 3-D Analytical Models of Fields

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

$$
\begin{aligned}
B_X &= \frac{\partial V}{\partial X}, \qquad B_Y = \frac{\partial V}{\partial Y}, \qquad B_Z = \frac{\partial V}{\partial Z} \\
\frac{\partial B_X}{\partial X} &= \frac{\partial^2 V}{\partial X^2}, \quad \frac{\partial B_X}{\partial Y} = \frac{\partial^2 V}{\partial X \partial Y}, \quad \text{etc.}
\end{aligned}
\tag{2.3.4}
$$

Multipoles

The scalar potential used for the calculation of the derivatives $\frac{\partial^{i+j+k}\vec{B}_{n,X,Y,Z}}{\partial X^i \partial Y^i \partial Z^k}$ $(i+j+k = 0 \text{ to } 4)$ for the magnetic and electromagnetic multipoles with $2n$ poles (namely, *QUADRUPO* $(n=2)$ to *DODECAPO* $(n=6)$, *MULTIPOL* $(n = 1 \text{ to } 6)$, *EBMULT* $(n = 1 \text{ to } 6)$) is [6]

$$
V_n(X,Y,Z) = (n!)^2 \left(\sum_{q=0}^{\infty} (-1)^q \frac{G^{(2q)}(X)(Y^2+Z^2)^q}{4^q q!(n+q)!}\right)\left(\sum_{m=0}^{n} \frac{sin\left(m\frac{\pi}{2}\right) Y^{n-m} Z^m}{m!(n-m)!}\right) \tag{2.3.5}
$$

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

$$G(s) = \frac{G_0}{1+\exp(P(s))}, \quad G_0 = \frac{B_0}{R_0^n} \tag{2.3.6}$$

and s is the distance to the EFB.

Skewed multipoles

Any multipole component n can be rotated independently by an 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, as derived from the expression (2.3.5) above. Second, $\vec{B}$ and its derivatives at (X, Y_R, Z_R) in the (X, Y, Z) frame are transformed to the rotated (X, Y_R, Z_R) frame by a rotation of the same angle A_n. A skewed $2n$-pole component is thus obtained by taking $A_n = \pi/2n$.

2.4 Calculation of $\vec{B}$ from Field Maps

2.4.1 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 a node i of the uniform mesh, which 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

$$B(X, 0) = A_0 + A_1 X + A_2 X^2 + A_3 X^3 + A_4 X^4 + A_5 X^5 \tag{2.4.1}$$

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

$$S = \sum_i (B(X, 0) - B_i)^2 \tag{2.4.2}$$

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 $\frac{\partial^n B}{\partial X^n}(X, 0)$ at the actual position X, as involved in eqs. (2.3.1), are then obtained by differentiation of the polynomial (2.4.1), giving

$$\begin{aligned}
\frac{\partial B}{\partial X}(X, 0) &= A_1 + 2A_2 X + 3A_3 X^2 + 4A_4 X^3 + 5A_5 X^4 \\
\frac{\partial^2 B}{\partial X^2}(X, 0) &= 2A_2 + 6A_3 X + 12A_4 X^2 + 20A_5 X^3 \\
&\cdots \\
\frac{\partial^5 B}{\partial X^5}(X, 0) &= 120A_5
\end{aligned} \tag{2.4.3}$$

2.4.2 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. (2.2.7).

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 a simulation, the grid simply aims at interpolating the field at a given point from its 9 or 25 neighbors. If the map results from measurements, the grid also smoothes field measurement 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

$$B(X,Y,0) = A_{00} + A_{10}X + A_{01}Y + A_{20}X^2 + A_{11}XY + A_{02}Y^2 \tag{2.4.4}$$

in second order, or

$$\begin{aligned} B(X,Y,0) =& A_{00} + A_{10}X + A_{01}Y + A_{20}X^2 + A_{11}XY + A_{02}Y^2 \\ &+ A_{30}X^3 + A_{21}X^2Y + A_{12}XY^2 + A_{03}Y^3 \\ &+ A_{40}X^4 + A_{31}X^3Y + A_{22}X^2Y^2 + A_{13}XY^3 + A_{04}Y^4 \end{aligned} \tag{2.4.5}$$

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

$$S = \sum_{ij} (B(X,Y,0) - B_{ij})^2 \tag{2.4.6}$$

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

$$A_{ij} = \frac{1}{i!j!} \frac{\partial^{i+j} B}{\partial X^i \partial Y^j} (0,0,0) \tag{2.4.7}$$

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 (2.4.4) in the case of second order interpolation)

$$\begin{aligned} \frac{\partial B}{\partial X}(X,Y,0) &= A_{10} + 2A_{20}X + A_{11}Y \\ \frac{\partial B}{\partial Y}(X,Y,0) &= A_{01} + A_{11}X + 2A_{02}Y \\ &\text{etc.} \end{aligned} \tag{2.4.8}$$

This allows stepping to the calculation of $\vec{B}(X,Y,Z)$ and its derivatives as described in subsection 2.3.2 (eq. 2.3.3).

The special case of polar maps

It is necessary to change from polar to Cartesian coordinates. This is done as follows.
In second order calculations the correspondence is

$$
\begin{aligned}
\frac{\partial B}{\partial X} &= \frac{1}{R}\frac{\partial B}{\partial \alpha} \\
\frac{\partial B}{\partial Y} &= \frac{\partial B}{\partial R} \\
\frac{\partial^2 B}{\partial X^2} &= \frac{1}{R^2}\frac{\partial^2 B}{\partial \alpha^2} + \frac{1}{R}\frac{\partial B}{\partial R} \\
\frac{\partial^2 B}{\partial X \partial Y} &= \frac{1}{R}\frac{\partial^2 B}{\partial \alpha \partial R} - \frac{1}{R^2}\frac{\partial B}{\partial \alpha} \\
\frac{\partial^2 B}{\partial Y^2} &= \frac{\partial^2 B}{\partial R^2} \\
\frac{\partial^3 B}{\partial X^3} &= \frac{3}{R^2}\frac{\partial^2 B}{\partial \alpha \partial R} - \frac{2}{R^3}\frac{\partial B}{\partial \alpha} \\
\frac{\partial^3 B}{\partial X^2 \partial Y} &= \frac{-2}{R^3}\frac{\partial^2 B}{\partial \alpha^2} - \frac{1}{R^2}\frac{\partial B}{\partial R} + \frac{1}{R}\frac{\partial^2 B}{\partial R^2} \\
\frac{\partial^3 B}{\partial X \partial Y^2} &= \frac{2}{R^3}\frac{\partial B}{\partial \alpha} - \frac{2}{R^2}\frac{\partial^2 B}{\partial \alpha \partial R} \\
\frac{\partial^3 B}{\partial Y^3} &= 0
\end{aligned}
\tag{2.4.9}
$$

In fourth order calculations the relations are the same up to second order, and then

$$
\begin{aligned}
\frac{\partial^3 B}{\partial X^3} &= \frac{1}{R^3}\frac{\partial^3 B}{\partial \alpha^3} + \frac{3}{R^2}\frac{\partial^2 B}{\partial \alpha \partial R} - \frac{2}{R^3}\frac{\partial B}{\partial \alpha} \\
\frac{\partial^3 B}{\partial X^2 \partial Y} &= \frac{1}{R^2}\frac{\partial^3 B}{\partial \alpha^2 \partial R} - \frac{2}{R^3}\frac{\partial^2 B}{\partial \alpha^2} - \frac{1}{R^2}\frac{\partial B}{\partial R} + \frac{1}{R}\frac{\partial^2 B}{\partial R^2} \\
\frac{\partial^3 B}{\partial X \partial Y^2} &= \frac{1}{R}\frac{\partial^3 B}{\partial \alpha \partial R^2} + \frac{2}{R^3}\frac{\partial B}{\partial \alpha} - \frac{2}{R^2}\frac{\partial^2 B}{\partial \alpha \partial R} \\
\frac{\partial^3 B}{\partial Y^3} &= \frac{\partial^3 B}{\partial R^3} \\
\frac{\partial^4 B}{\partial X^4} &= \frac{1}{R^4}\frac{\partial^4 B}{\partial \alpha^4} - \frac{8}{R^4}\frac{\partial^2 B}{\partial \alpha^2} + \frac{6}{R^3}\frac{\partial^3 B}{\partial \alpha^2 \partial R} + \frac{3}{R^2}\frac{\partial^2 B}{\partial R^2} - \frac{3}{R^3}\frac{\partial B}{\partial R} \\
\frac{\partial^4 B}{\partial X^3 \partial Y} &= \frac{1}{R^3}\frac{\partial^4 B}{\partial \alpha^3 \partial R} - \frac{3}{R^4}\frac{\partial^3 B}{\partial \alpha^3} + \frac{3}{R^2}\frac{\partial^3 B}{\partial \alpha \partial R^2} - \frac{8}{R^3}\frac{\partial^2 B}{\partial \alpha \partial R} + \frac{6}{R^4}\frac{\partial B}{\partial \alpha} \\
\frac{\partial^4 B}{\partial X^2 2Y^2} &= \frac{1}{R^4}\frac{\partial^2 B}{\partial \alpha^2} - \frac{4}{R^3}\frac{\partial^3 B}{\partial \alpha^2 \partial R} - \frac{2}{R^2}\frac{\partial^2 B}{\partial R^2} + \frac{2}{R^3}\frac{\partial B}{\partial R} + \frac{1}{R^2}\frac{\partial^4 B}{\partial \alpha^2 \partial R^2} + \frac{1}{R}\frac{\partial^3 B}{\partial R^3} \\
\frac{\partial^4 B}{\partial X \partial Y^3} &= \frac{1}{R}\frac{\partial^4 B}{\partial \alpha \partial R^3} - \frac{3}{R^2}\frac{\partial^3 B}{\partial \alpha \partial R^2} + \frac{6}{R^3}\frac{\partial^2 B}{\partial \alpha \partial R} - \frac{6}{R^4}\frac{\partial^4 B}{\partial \alpha^4} \\
\frac{\partial^4 B}{\partial Y^4} &= \frac{\partial^4 B}{\partial R^4}
\end{aligned}
\tag{2.4.10}
$$

NOTE: If a particle goes beyond the limits of the field map, the field and its derivatives will be extrapolated by means of the same calculations, from the border grid which is the closest to the actual position of the particle. Its flag *IEX* is given the value -1 (see section 5.6.8).

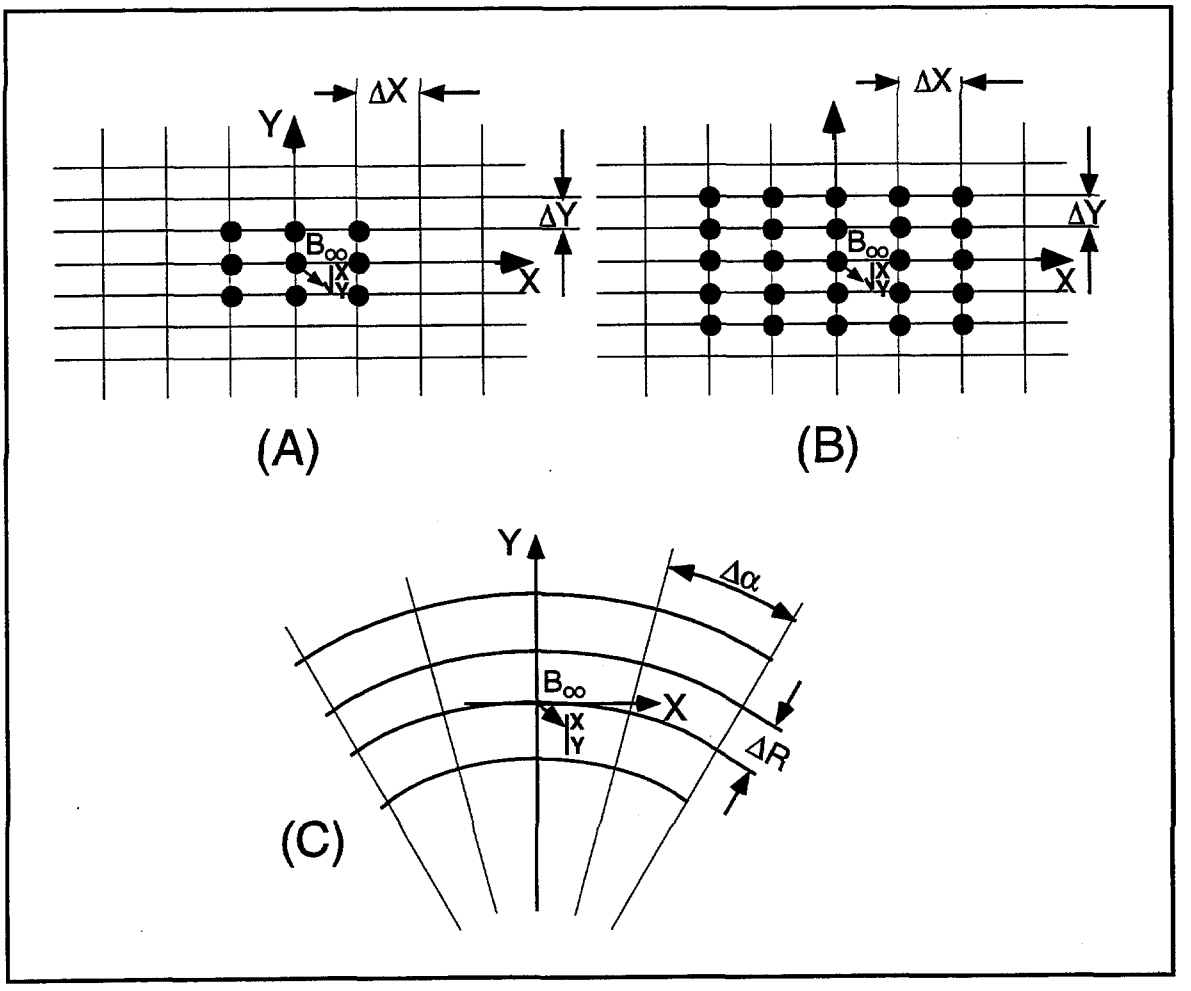


Figure 3: Mesh in the (X, Y) plane in Cartesian coordinates. The grid is centered
on the node which is closest to the actual position of the particle.
A: 9-point interpolation grid.
B: 25-point interpolation grid.
C: Mesh in the (X, Y) plane in polar coordinates.

2.4.3 Arbitrary 2-D Map, no Symmetries

The map is supposed 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 a polynomial interpolation from a 3×3 points grid centered at the node (i, j) which is closest to the position (X, Y)

$$B_\ell(X, Y, Z_0) = A_{00} + A_{10}X + A_{01}Y + A_{20}X^2 + A_{11}XY + A_{02}Y^2 \tag{2.4.11}$$

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

$$\frac{\partial B_\ell}{\partial X}(X,Y,Z_0) = A_{10} + 2A_{20}X + A_{11}Y$$
$$\frac{\partial^2 B_\ell}{\partial X \partial Y}(X,Y,Z_0) = A_{11} \qquad (2.4.12)$$
etc.

Then follows the procedure of extrapolation from (X,Y,Z_0) to the actual position (X,Y,Z), as described in section 2.3.3

No special symmetries are assumed, which allows the treatment of any type of magnet.

2.4.4 Calculation of $\vec{B}$ from 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 now defined by means of a 3-D field map, through second order polynomial interpolation

$$B_\ell(X,Y,Z) = A_{000} + A_{100}X + A_{010}Y + A_{001}Z + A_{200}X^2 + A_{020}Y^2 + A_{002}Z^2 + A_{110}XY + A_{101}XZ + A_{011}YZ \qquad (2.4.13)$$

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

$$\frac{\partial B_\ell}{\partial X} = A_{100} + 2A_{200}X + A_{110}Y + A_{101}Z$$
$$\frac{\partial^2 B_\ell}{\partial X^2} = 2A_{200} \qquad (2.4.14)$$

and so on for first and second order derivatives with respect to X, Y or Z.

The interpolation involves a $3 \times 3 \times 3$-point parallelipipedic 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

$$S = \sum_{ijk} \left(B_\ell(X,Y,Z) - B^\ell_{ijk}\right)^2 \qquad (2.4.15)$$

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 $\partial S/\partial A_{ijk} = 0$.

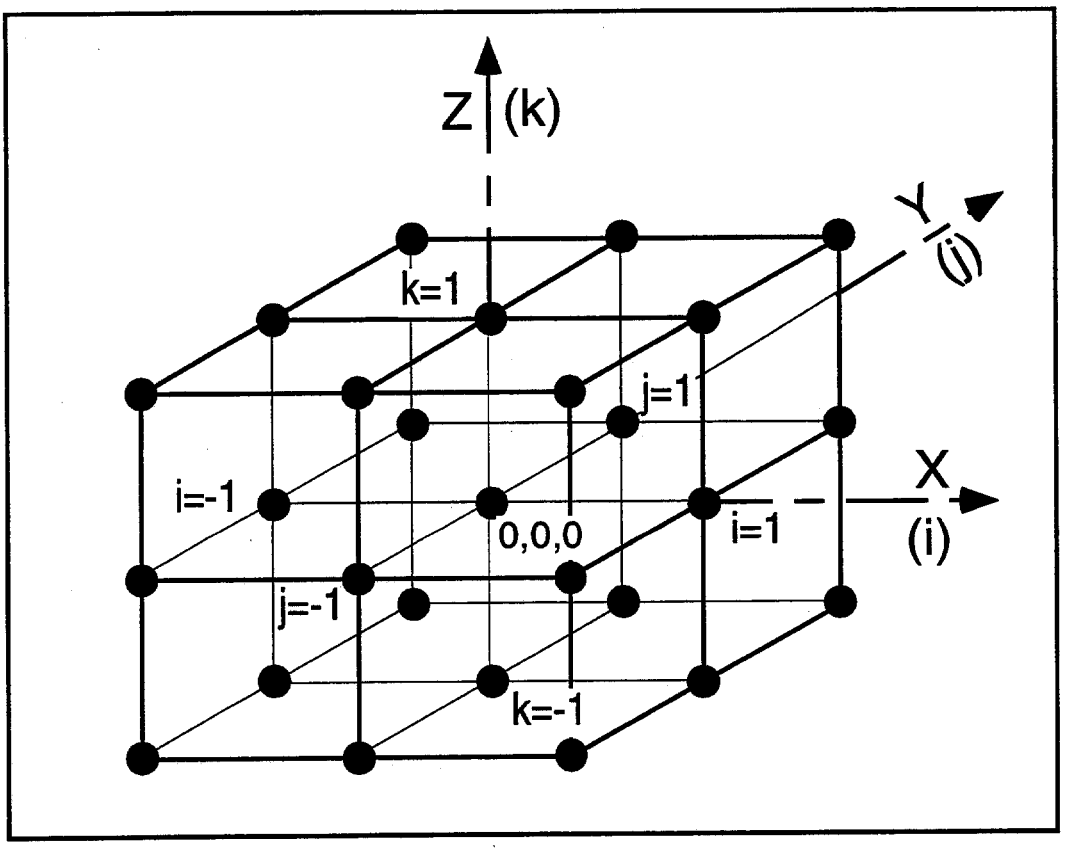


Figure 4: A 3-D 27-point grid is used for interpolation of $\vec{B}$ and its derivatives up to second order. The central node of the grid ($i = j = k = 0$) is at the closest vicinity of the actual position of the particle.

2.5 Calculation of $\vec{E}$ and its derivatives

Zgoubi calculates $\vec{E}(X,Y,Z)$ and its derivatives in several different ways, depending on whether field maps or analytical representations of optical elements are used. The three basic means are the following [7].

2.5.1 Extrapolation from a 1-D axial field map

A cylindrically symmetric field can be described by an axial 1-D field map of its longitudinal component $E_X(X, r = 0)$ ($r = (Y^2 + Z^2)^{1/2}$), while the radial component $E_r(X, r = 0)$ is assumed to be zero (e.g. in *ELREVOL*). $E_X(X, r = 0)$ is obtained at any point along the X-axis by a polynomial interpolation from the map mesh (see section 2.4.1). Then the field components $E_X(X,r)$, $E_r(X,r)$ at the position of the particle, (X,r) are obtained from Taylor expansions to the fifth order in r (hence, up to the fifth order derivative $\frac{\partial^5 E_X}{\partial X^5}(X,0)$), assuming cylindrical symmetry

$$\begin{aligned} E_X(X,r) &= E_X(X,0) - \frac{r^2}{4}\frac{\partial^2 E_X}{\partial X^2}(X,0) + \frac{r^4}{64}\frac{\partial^4 E_X}{\partial X^4}(X,0) \\ E_r(X,r) &= -\frac{r}{2}\frac{\partial E_X}{\partial X}(X,0) + \frac{r^3}{16}\frac{\partial^3 E_X}{\partial X^3}(X,0) - \frac{r^5}{384}\frac{\partial^5 E_X}{\partial X^5}(X,0) \end{aligned} \tag{2.5.1}$$

By differentiation with respect to X and r, up to the second order, these expressions provide the derivatives of $\vec{E}(X,r)$. Finally a conversion from the (X,r) coordinates to the (X,Y,Z) Cartesian coordinates of **Zgoubi** is performed, thus providing the expressions $\frac{\partial^{i+j+k}\vec{E}}{\partial X^i \partial Y^j \partial Z^k}$ needed in the eqs. (2.2.12).

2.5.2 Extrapolation from analytically defined axial fields

This procedure assumes cylindrical symmetry with respect to the X-axis. The longitudinal field component $E_X(X,r=0)$ ($r=(Y^2+Z^2)^{1/2}$), along this axis are derived from differentiation of an adequate model of the electric potential $V(X)$ (e.g. in *EL2TUB, UNIPOT*). The longitudinal and radial field components $E_X(X,r)$, $E_r(X,r)$ and their derivatives off-axis $\frac{\partial^{i+j}E_X}{\partial X^i \partial r^j}$ and $\frac{\partial^{i+j}E_r}{\partial X^i \partial r^j}$ are obtained by Taylor expansions to the fifth order in r assuming cylindrical symmetry (see eq. (2.5.1)), and then transformed to the (X,Y,Z) Cartesian frame of **Zgoubi** in order to provide the derivatives $\frac{\partial^{i+j+k}\vec{E}}{\partial X^i \partial Y^j \partial Z^k}$ needed in eq. (2.2.12).

2.5.3 3-D Analytical models of fields

In analytical elements (e.g. *WIENFILT, ELMULT, EBMULT*), the three components of $\vec{E}$, namely E_X, E_Y, E_Z, and their derivatives with respect to X, Y or Z are derived at any step along trajectories, from the analytical expressions of field models that give $\vec{E}(X,Y,Z)$.

Multipoles and skewed multipoles

A right electric multipole is considered to have the same effect as the equivalent skewed magnetic multipole. Therefore, the calculation of the right electric or electromagnetic multipoles (*ELMULT, EBMULT*) uses the same eq. (2.3.5) together with the rotated process described in section 2.3.5. The same method is used, for rotating arbitrary multipole components around the X-axis, whatever the angle of rotation.

2.6 Calculation of $\vec{E}$ from field maps

1-D axial map, with cylindrical symmetry

The only type of field map treated in the actual version is the 1-D axial map, with cylindrical symmetry. The same procedure as for the case of magnetic fields is involved (see section 2.4.1).

3 SPIN TRACKING [8]

The depolarization of a particle beam travelling in a magnetic field $\vec{B}$ takes its origin in the spin precession undergone by each particle. This motion of the spin $\vec{S}$ is governed by the Thomas-BMT first order differential equation [9]

$$\frac{d\vec{S}}{dt} = \frac{q}{\gamma m}\vec{S}\times\vec{\Omega} \tag{3.1}$$

where

$$\vec{\Omega} = (1+\gamma G)\vec{b} + G(1-\gamma)\vec{b}_{//} \tag{3.2}$$

q, m, γ and G are respectively the charge, rest mass, Lorentz relativistic factor, and anomalous magnetic moment of the particle. $\vec{b}_{//}$ is the component of $\vec{b}$ which is parallel to the velocity $\vec{v}$ of the particle.
These equations are normalized by introducing the same notation as previously. Let $b = \| \vec{b} \|$ and $v = \| \vec{v} \|$; $ds = vdt$ is the differential path, $\frac{\gamma m v}{q} = B\rho$ is the rigidity of the particle; $\vec{S}' = \frac{d\vec{S}}{ds} = \frac{1}{v}\frac{d\vec{S}}{dt}$ is the derivative of the spin with respect to the path.

Introducing also $\vec{B} = \dfrac{\vec{b}}{B\rho}$ and $\vec{B}_{/\!/} = \dfrac{\vec{b}_{/\!/}}{B\rho}$, and

$$\vec{\omega} = \frac{\vec{\Omega}}{B\rho} = (1+\gamma G)\vec{B} + G(1-\gamma)\vec{B}_{/\!/} \tag{3.3}$$

eq. (3.1) can be re-written in a normalized way

$$\vec{S}' = \vec{S} \times \vec{\omega} \tag{3.4}$$

This equation is then solved in the same way as the reduced Lorentz equation (2.2.3). From the values of the magnetic factor $\vec{\omega}(M_0)$ and the spin $\vec{S}(M_0)$ of the particle at position M_0 of its trajectory, the spin $\vec{S}(M_1)$ at position M_1, following a displacement ds (fig. 2), is given by the Taylor expansion

$$\vec{S}(M_1) = \vec{S}(M_0) + \frac{d\vec{S}}{ds}(M_0)\,ds + \frac{d^2\vec{S}}{ds^2}(M_0)\frac{ds^2}{2} + \frac{d^3\vec{S}}{ds^3}(M_0)\frac{ds^3}{3!} + \frac{d^4\vec{S}}{ds^4}(M_0)\frac{ds^4}{4!} \tag{3.5}$$

The derivatives $\vec{S}^{(n)} = \dfrac{d^n\vec{S}}{ds^n}$ of $\vec{S}$ at M_0 are obtained by differentiating eq. (3.4)

$$\begin{aligned}
\vec{S}' &= \vec{S} \times \vec{\omega} \\
\vec{S}'' &= \vec{S}' \times \vec{\omega} + \vec{S} \times \vec{\omega}' \\
\vec{S}''' &= \vec{S}'' \times \vec{\omega} + 2\vec{S}' \times \vec{\omega}' + \vec{S} \times \vec{\omega}'' \\
\vec{S}'''' &= \vec{S}''' \times \vec{\omega} + 3\vec{S}'' \times \vec{\omega}' + 3\vec{S}' \times \vec{\omega}'' + \vec{S} \times \vec{\omega}'''
\end{aligned} \tag{3.6}$$

where the derivatives $\vec{\omega}^{(n)}$ are obtained from eq. (3.3).

The last point consists in getting $\vec{B}_{/\!/}$ and its derivatives. This can be done in the following way. Let $\vec{u} = \dfrac{\vec{v}}{v}$ be the normalized velocity of the particle, then,

$$\begin{aligned}
\vec{B}_{/\!/} &= (\vec{B}\cdot\vec{u})\,\vec{u} \\
\vec{B}'_{/\!/} &= (\vec{B}'\cdot\vec{u} + \vec{B}\cdot\vec{u}')\,\vec{u} + (\vec{B}\cdot\vec{u})\,\vec{u}' \\
\vec{B}''_{/\!/} &= (\vec{B}''\cdot\vec{u} + 2\vec{B}'\cdot\vec{u}' + \vec{B}\cdot\vec{u}'')\,\vec{u} + 2(\vec{B}'\cdot\vec{u} + \vec{B}\cdot\vec{u}')\,\vec{u}' + (\vec{B}\cdot\vec{u})\,\vec{u}'' \\
&\text{etc.}
\end{aligned} \tag{3.7}$$

The quantities $\vec{u}$, $\vec{B}$ and their n-th derivatives as involved in these equations are picked up from eqs. (2.2.6, 2.2.7).

4 SYNCHROTRON RADIATION [10]

The ray-tracing procedures[1] provide the ingredients necessary for the determination of the electric field radiated by the particle subject to acceleration, as shown in Fig. 5.

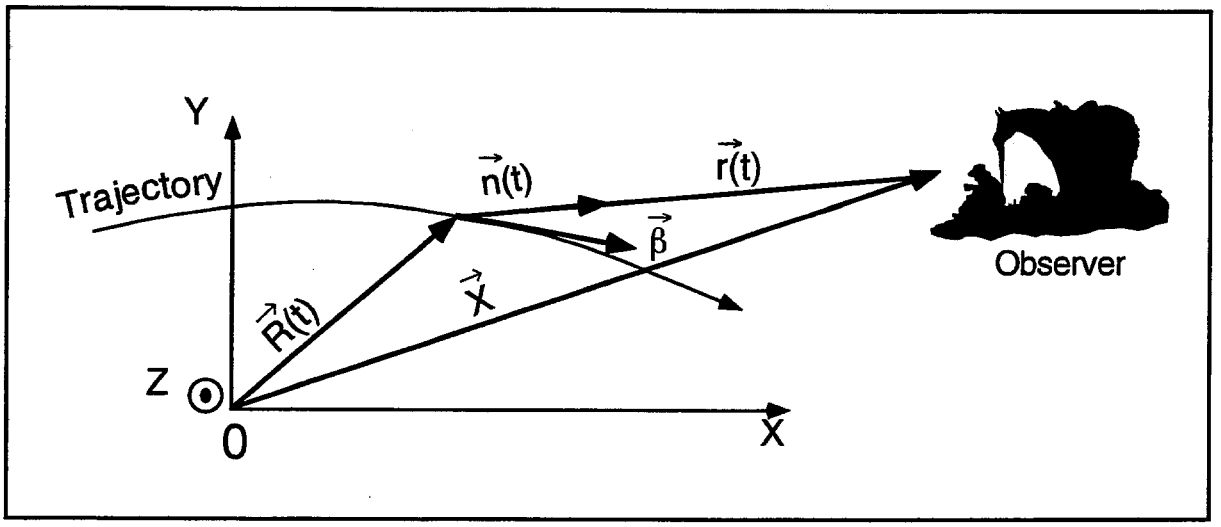


Figure 5: A scheme of the reference frame in **Zgoubi** together with the vectors entering in the definition of the electric field radiated by the accelerated particle:
(x, y): horizontal plane; z: vertical axis.
$\vec{R}(t)$ = particle position in the fixed frame (O, x, y, z);
$\vec{X}$ (time-independent) = 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 $\vec{v}/c = (1/c)d\vec{R}/dt$.

4.1 Calculation of the electric field $\mathcal{E}(\vec{n}, \tau)$

The expression for $\mathcal{E}(\vec{n}, \tau)$ as seen by the observer in the long distance approximation is [11]

$$\mathcal{E}(\vec{n}, \tau) = \frac{q}{4\pi\varepsilon_0 c} \frac{\vec{n}(t) \times \left[\left(\vec{n}(t) - \vec{\beta}(t) \right) \times d\vec{\beta}/dt \right]}{r(t) \left(1 - \vec{n}(t) \cdot \vec{\beta}(t) \right)^3} \tag{4.1.1}$$

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 differential relation

[1] Also implemented in the post-processor **zgplot**

$$d\tau/dt = 1 - \vec{n}(t) \cdot \vec{\beta}(t) \tag{4.1.2}$$

The vectors $\vec{R}(t)$ and $\vec{\beta}(t) = \frac{v}{c}\vec{u}$ (Eq. 2.2.2) that describe the motion are obtained from the ray-tracing (Eqs. 2.2.4). The acceleration is calculated from (Eq. 2.2.1)

$$d\vec{\beta}/dt = (q/m)\ \vec{\beta}(t) \times \vec{b}(t) \tag{4.1.3}$$

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

$$\vec{r}(t) = \vec{X} - R(t) \text{ and } \vec{n}(t) = \vec{r}(t)/|\vec{r}(t)| \tag{4.1.4}$$

The calculation 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

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

since the preferred direction of observation is generally almost parallel to $\vec{\beta}$ (exactly 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 software form for achieving accurate computation.
The expression for $\vec{n}$ is

$$\begin{aligned}\vec{n} = (n_x, n_y, n_z) &= (\cos\Psi\cos\phi, \cos\Psi\sin\phi, \sin\Psi) \\ &= \left[1 - 2(\sin^2\phi/2 + \sin^2\Psi/2) + 4\sin^2\phi/2\sin^2\Psi/2, \sin\phi(1 - 2\sin^2\Psi/2), \sin\Psi\right]\end{aligned} \tag{4.1.5}$$

where ϕ and Ψ are the observation angles, given by

$$\phi = \text{Atg}\left(\frac{r_y}{r_x}\right) \text{ and } \Psi = \text{Atg}\left(\frac{r_z}{\sqrt{r_x^2 + r_y^2}}\right) \tag{4.1.6}$$

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

$$\begin{aligned}\vec{\beta} = (\beta_x, \beta_y, \beta_z) &= \left[\sqrt{(\beta^2 - \beta_y^2 - \beta_z^2)}, \beta_y, \beta_z\right] \\ &= \left[\sqrt{(1 - 1/\gamma^2 - \beta_y^2 - \beta_z^2)}, \beta_y, \beta_z\right] = (1 - a/2 + a^2/8 - a^3/16 + ..., \beta_y, \beta_z)\end{aligned} \tag{4.1.7}$$

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

$$\vec{n}_x = 1 - \varepsilon_x \text{ and } \vec{\beta}_x = 1 - \xi_x$$

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,

$$\vec{n} - \vec{\beta} = (-\varepsilon_x + \xi_x, n_y - \beta_y, n_z - \beta_z) \ , \tag{4.1.8}$$

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,

$$1 - \vec{n} \cdot \vec{\beta} = \varepsilon_x + \xi_x - n_y\beta_y - n_z\beta_z - \varepsilon_x\xi_x \ , \tag{4.1.9}$$

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 $a \sim 1/\gamma^2 << 1$.

4.2 Calculation of the Fourier transform of the electric field

The Fourier transforms

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

of the σ and π electric field components provide the spectral angular brightness

$$\partial^3 P / \partial\phi \, \partial\Psi \, \partial\omega = 2r^2 \left| FT_\omega \left(\mathcal{E}(\tau) \right) \right|^2 / \mu_0 c \tag{4.2.1}$$

They are calculated in a regular way, without using the FFT technic, namely from

$$FT_\omega \left[\mathcal{E}(\tau) \right] \approx \sum \mathcal{E}(\tau_n) \, e^{-i\omega\tau_n} \, \Delta\tau_n \tag{4.2.2}$$

for two reasons. On the one hand, the number of integration steps ds that define the trajectory (Eqs. 2.2.4), is fully arbitrary and therefore in general not of order 2^n. On the other hand, the integration step defines a constant time differential element $\Delta t_n = ds/\beta c$ which results in the observer differential time element $\Delta\tau_n$, which is also the differential element of the Fourier transform, being non-constant, since both are related by eq. 4.1.2 in which $\vec{\beta}$ and $\vec{n}$ vary as a function of the number n of integration steps.
Another major point is that $\Delta\tau_n$ may reach drastically small values in the region of the central peak of the electric impulse emitted in a dipole ($1 - \vec{n}(t) \cdot \vec{\beta}(t) \to 1/2\gamma^2$), while the total integrated time $\sum \Delta\tau_n$ 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 $\sum \tau_n$ 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 $2\tau_c = \dfrac{2(1+\gamma^2\psi^2)^{3/2} \, 2\rho}{3\gamma^3 \, c}$ of the peak of radiation [12], is a very small fraction of $\sum \tau_n$.
The consequence is that, once again in relation with computer precision, the differential element $\Delta\tau_n$ involved in the computation of eq. 4.2.2 cannot be derived from such relation as $\Delta\tau_n = \tau_n - \tau_{n-1}$, but instead must be stored as such beforehand.

5 DESCRIPTION OF THE AVAILABLE PROCEDURES

5.1 *Introduction*

This chapter gives a detailed description of how the **Zgoubi** procedures work, and their associated keywords. It has been split into several sections. Sections 5.2 to 5.5 explain the underlying content and functioning of all available keywords. Section 5.6 is dedicated to the description of some general procedures that may be accessed by means of special data or flags (such as negative integration steps), or through the available keywords (such as multiturn tracking with *REBELOTE*).

5.2 Definition of an Object

The description of the object, *i.e.*, initial coordinates of the beam, must be the first element of the input data to **Zgoubi**.

Several types of automatically generated objects are available, as described in the following pages.

MCOBJET: - Monte Carlo Generation of a 6-D Object

MCOBJET generates a set of up to 200 random initial conditions. It is generally used in conjunction with the keyword *REBELOTE*, which allows generating an arbitrarily high number of initial conditions.

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

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

Depending on the value of the next datum, *KOBJ*, the *IMAX* (≤ 200) particles have their initial random conditions Y, T, Z, P, X and D (relative momentum) generated on 3 different types of supports, as described below.
Next come the data

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

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

1. uniform density, $p(x) = 1$ 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, $p(D) = 1$ 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 value 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 5.6.9).

KOBJ = 1: Random generation of *IMAX* particles in a hyper-window with widths (i.e., the half-extents 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* and *MCDESINT*).

KOBJ = 2: Random generation of $IY * IT * IZ * IP * IX * ID$ particles (maximum 200) in 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. 6.

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 inside a 6-D ellipsoid defined by the three sets of data (one set per 2-D phase-space)

$$\begin{array}{llll} \alpha_Y, & \beta_Y, & \frac{\varepsilon}{\pi}\varepsilon_Y, & N_{\varepsilon Y}[, \quad N'_{\varepsilon Y}, \text{ if } N_{\varepsilon Y} < 0] \\ \alpha_Z, & \beta_Z, & \frac{\varepsilon}{\pi}\varepsilon_Z, & N_{\varepsilon Z}[, \quad N'_{\varepsilon Z}, \text{ if } N_{\varepsilon Z} < 0] \\ \alpha_X, & \beta_X, & \frac{\varepsilon}{\pi}\varepsilon_X, & N_{\varepsilon X}[, \quad N'_{\varepsilon X}, \text{ if } N_{\varepsilon X} < 0] \end{array}$$

where α, β are the ellipse parameters and ε/π the emittance, corresponding to a frontier given, e.g., in the (Y,T) plane by $\frac{1+\alpha_Y^2}{\beta_Y}Y^2 + 2\alpha_Y YT + \beta_Y T^2 = \varepsilon_Y/\pi$ (idem for the (Z,P) or (X,D) planes). $N_{\varepsilon Y}$, $N_{\varepsilon Z}$ and $N_{\varepsilon X}$ are the sorting cut-offs (used only for Gaussian distributions, $KY, KT, ... = 2$).
The sorting is uniform in surface (for $KY = 1$, or $KZ = 1$ or $KX = 1$) or Gaussian ($KY = 2$ or $KZ = 2$), and so on, as described above. A uniform sorting has the ellipse above for support. A Gaussian sorting has the ellipse above for r.m.s. frontier, leading to $\sigma_Y = \sqrt{\beta_Y \varepsilon_Y/\pi}$, $\sigma_T = \sqrt{\frac{(1+\alpha_Y^2)}{\beta_Y}\varepsilon_Y/\pi}$, and similar relations for σ_Z, σ_X.
If N_ε is negative, thus the sorting fills the elliptical ring that extends from $|N_\varepsilon|$ to N'_ε (rather than the inner region determined by the N_ε cut-off, as addressed above).

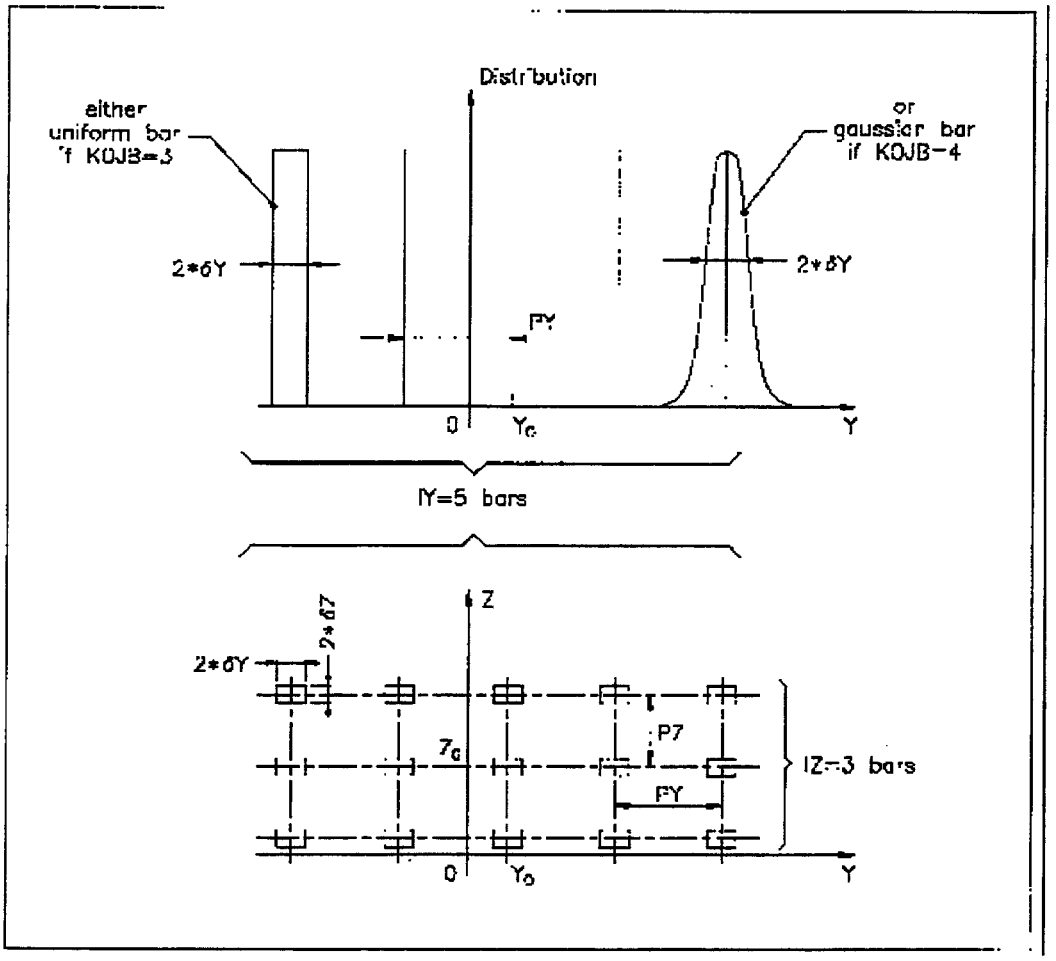


Figure 6: Scheme of the input parameters to *MCOBJET* when $KOBJ = 3, 4$
A: A distribution of the Y coordinate
B: 2-D grid in (Y, Z) space.

OBJET: - Generation of an Object

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

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

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

At the object, the beam is defined by a set of particles (maximum 200) with the initial conditions (Y, T, Z, P, X, D) where D is the relative momentum.
Depending on the value of the next datum *KOBJ*, these initial conditions may be generated in six different ways:

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

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

(maximum 21 in each coordinate: $IY \leq 21 \ldots ID \leq 21$) and the sampling size

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

Zgoubi then generates $IY * IT * IZ * IP * IX * ID$ (≤ 200) initial conditions with the following coordinates

$$\begin{array}{llll} 0, & \pm PY, & \pm 2 * PY, & \ldots, \quad \pm IY/2 * PY, \\ 0, & \pm PT, & \pm 2 * PT, & \ldots, \quad \pm IT/2 * PT, \\ 0, & \pm PZ, & \pm 2 * PZ, & \ldots, \quad \pm IZ/2 * PZ, \\ 0, & \pm PP, & \pm 2 * PP, & \ldots, \quad \pm IP/2 * PP, \\ 0, & \pm PX, & \pm 2 * PX, & \ldots, \quad \pm IX/2 * PX, \\ 0, & \pm PD, & \pm 2 * PD, & \ldots, \quad \pm ID/2 * PD, \end{array}$$

In this option relative momenta will be classified automatically for the purpose of the use of *IMAGES* for momentum analysis.
The particles are tagged with an index *IREP* eventually 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 of them 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 the relative momentum of the problem, $\mathcal{D}$: the reference rigidity of the beam is $\mathcal{D} * BORO$, resulting in the rigidity of a particle of initial condition $I * PD$, for instance, to be $(\mathcal{D} + I * PD) * BORO$.

KOBJ = 2: Next data: *IMAX*, *IDMAX*. Initial coordinates are entered explicitly for each trajectory. *IMAX* is the total number of particles ($IMAX \leq 200$). These may be classified in groups of equal number for each value of momentum, in order to fit the requirements of image calculations by *IMAGES*. *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* $\times 1$ (which indicates that the particle will be tracked) or -2 (indicates that the particle will not be tracked).
This option $KOBJ = 2$ may be be useful for the definition of objects including kinematic effects.

KOBJ $= \pm 3$**:** Next data: *IMAX*, *IDMAX* as explained for $KOBJ = 2$.
This option allows the reading of initial conditions from an external input file. This file must be formatted so as to fit the following *FORTRAN* sequence

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD')
DO 1 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),
100       FORMAT (1X, A1, 1X, I2, 6E16.8, / , 6E16.8, 2I3, / )
  1     CONTINUE
```

where the meaning of the parameters is the following

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

IREP is an index which indicates a symmetry with respect to 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 deduced from ray-tracing but instead from those of particle I by simple symmetry. This results in gain of computing time.
If $KOBJ = +3$, further ray-tracing starts from the current coordinates $F(J, I)$. If $KOBJ = -3$, further ray-tracing starts from the initial coordinates $FO(J, I)$.

$KOBJ = \pm 3$ can be used directly for reading files filled by *FAISCNL*.
If more than 200 particles are to be read from a file, use $IMAX \leq 200$ in conjunction with *REBELOTE*.

KOBJ $= 4$: Same as $KOBJ = 1$ except for the Z symmetry. The initial Z and P conditions are the following

$$0, \quad \pm PZ, \quad \pm 2 * PZ, \quad \ldots, \quad \pm(IZ-1) * PZ,$$
$$0, \quad \pm PP, \quad \pm 2 * PP, \quad \ldots, \quad \pm(IP-1) * PP,$$

This object results in shorter outputs when studying problems with Z symmetry.

KOBJ $= 5$**:** Mostly dedicated to the calculation of first order transfer matrices, in conjunction with *MATRIX*. The input data are the step sizes

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

The code generates 11 particles

$$0, \quad \pm PY, \quad \pm PT, \quad \pm PZ, \quad \pm PP, \quad \pm PX, \quad \pm PD$$

These values should be small enough, so that the paraxial ray approximation be valid.
The last data are the initial coordinates of the reference trajectory [normally $(YR, TR, ZR, PR, XR, DR) = (0, 0, 0, 0, 0, 1)$]. The reference rigidity is $\mathcal{DR} * BORO$ (negative value allowed).

KOBJ $= 6$**:** Mostly dedicated to the calculation of first, second and higher order transfer coefficients, in conjunction with *MATRIX*. The input data are the step sizes

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

to allow the building up of an object containing 61 particles. The last data are the initial coordinates of the reference trajectory [normally $(YR, TR, ZR, PR, XR, DR) = (0, 0, 0, 0, 0, 1)$]. The reference rigidity of the beam is $\mathcal{DR} * BORO$.

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 momentum D_I is calculated from the initial angle T_I following

$$D_I = \mathcal{D} + PD * T_I$$

while T_I is in the range

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

as stated under $KOBJ = 1$

OBJETA: - Object From Monte Carlo Simulation of Decay Reaction [13]

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:

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

The first input data are the reference rigidity

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

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*, and 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 ($\simeq 10^6$).

5.3 Declaration of options

These options allow the control of procedures that affect certain functions of the code. Some options are normally declared right after the object definition (e.g. *SPNTRK* - spin tracking, *MCDESINT* - in-flight decay), - the others are normally declared at the end of the data pile (e.g. *END* – end of a problem, *REBELOTE* – for tracking more than 200 particles, *FIT* – fitting procedure).

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, next line, by NF (≤ 20), the number of files to be translated.
Then follow, line per line, the NF names of the files to be translated.
Iff a file name begins with the prefix "B_", it is presumed "binary", and hence converted to "formatted", and given the same name after suppression of the prefix "B_". Conversely, iff the file name does not begin with "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* supports only files with the standard *TOSCA* magnet code output format (see the keyword *TOSCA*).

FIN or END : End of Input Data List

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

Any information following these keywords will be ignored.

FIT: Fitting Procedure

The keyword *FIT* allows the automatic adjustment of up to 20 variables, for fitting up to 20 constraints. It has been realized after existing routines used in the matrix transport code BETA [14]. Any physical parameter of any element (*i.e.* keyword) may be varied. Available constraints are: any of the 6×6 coefficients of the first order transfer matrix $[R_{ij}]$ as defined in the keyword *MATRIX*, and its horizontal $(R_{11}R_{22} - R_{12}R_{21})$ and vertical $(R_{33}R_{44} - R_{34}R_{43})$ determinants; any of the $6 \times 6 \times 6$ coefficients of the second order array $[T_{ijk}]$ as defined in *MATRIX* ; any of the 2×4 coefficients of the beam matrix as defined by

$$[\sigma_{ij}] = \begin{pmatrix} \sigma_{11} & \sigma_{12} & & \\ \sigma_{21} & \sigma_{22} & & \\ & & \sigma_{33} & \sigma_{34} \\ & & \sigma_{43} & \sigma_{44} \end{pmatrix}$$

and any trajectory coordinates $F(J, I)$ as defined in *OBJET* (I = particle number, J = coordinate number = 1 to 6 for respectively D, Y, T, Z,P or S =path length).

VARIABLES

The first input data in *FIT* are the number of variables *NV*, and for each one of them, the following parameters

- $IR =$ 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).
- $DV =$ allowed relative range of variation of the physical parameter IP.

Numbering of the elements (*IR*):

The elements (*DIPOLE, QUADRUPO,* etc.) are numbered following their sequence in the **Zgoubi** input data file, for the purpose of the *FIT* procedure. The number of any element just identifies to its position in the data sequence. However, a simple way to get IR is to make a preliminary run: **Zgoubi** will then print the whole structure in zgoubi.res with all elements numbered.

Numbering of the physical parameters (*IP*):

In the elements *DIPOLE*, *AIMANT* and *EBMULT*, *ELMULT*, *MULTIPOL*, the numbering of the physical parameters just follows their sequence, as it is shown here after for *DIPOLE*: the left column below represents the input data, the right one the corresponding numbering to be used for the *FIT* procedure.

Input data	**Numbering for FIT**
DIPOLE	
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.	etc.

For all other keywords, the parameters are numbered in the following way

KEYWORD	
first line	1, 2, 3,...
second line	10, 11, 12, 13,...
this is a comment	a line of comments is skipped
next line	20, 21, 22,...
and so on...	30, 31, 32, 33,...

The examples of *QUADRUPO* (quadrupole) and *TOSCA* (Cartesian mesh field map) are given below.

Input data	**Numbering for FIT**
MULTIPOL	
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, S_2, S_3, S_4, S_5, S_6	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	10
TIT	This is text
IX, IY, IZ	20, 21
FNAME	This is text
ID, A, B, C [A', B', C', etc if $ID \geq 2$]	30, 31, 32, 33 [34, 35, 36, etc if $ID \geq 2$]
IORDRE	40
XPAS	50
KPOS, XCE, YCE, ALE	60,61,62,63

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 $r \cdot pp$ where the integer part r is the number of the coupled element in the structure (equivalent to IR, see above), and the decimal part pp is the number of its parameter of concern (equivalent to IP, see above) (if the parameter number is in the range 1,...,9, then pp must take the form $0p$). For example, $XC = 20 \cdot 01$ is a request for coupling with the parameter number 1 of element number 20 of the structure, while $XC = 20 \cdot 10$ is a request for coupling with the parameter number 10 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* 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 of a symmetric 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 (DV)

For a parameter IP of initial value p, the *FIT* procedure is allowed to explore the range $p(1 \pm DV)$.

CONSTRAINTS

The next input data in *FIT* are the number of constraints, *NC*, and for each one of them the following parameters.

IC	=	type of the constraint (see table below).
I, J	=	constraint (*i.e.* R_{ij}, or determinants; T_{ijk}; σ_{ij}; trajectory and coordinate numbers)
IR	=	number of the element in the **Zgoubi** input data file, right after which the constraint applies
V	=	desired value of the constraint
W	=	weight of the constraint (smaller W for higher weight)

Type of constraint	**Parameters defining the constraint**			
	IC	**I**	**J**	**Constraint**
Beam matrix	0	1 - 4	1 - 4	σ_{IJ}
First order transfer coefficients		1 - 6	1 - 6	R_{IJ}
	1	7	any	Horizontal determinant
		8	any	Vertical determinant
Second order transfer coefficients	2	1 - 6	11 - 66	$T_{I,j,k}$ $(j = [J/10], k = J - 10[J/10])$
Trajectory coordinate	3	1 - IMAX	1 - 6	$F(J, I)$

Table 1: This table shows the constraints available, depending on the values of IC, I and J. [ ] denotes the integer part. When $IC = 3$, I designates the particle number and J the coordinate number (i.e., D, Y, T, Z, P or X).

The coefficients $\sigma_{11}(\sigma_{33})$ = horizontal (vertical) dimension of the beam, and $\sigma_{22}(\sigma_{44})$ = horizontal (vertical) divergence of the beam are calculated by means of the procedures described in *IMAGE*.
The fitting of the $[\sigma_{ij}]$ matrix coefficients supposes the tracking of a relevant population of particles within an adequate emittance.

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 or determinants supposes the tracking of particles having initial coordinates sampled as described in *MATRIX* (these particles are normally defined with *OBJET*, $KOBJ = 5$ or 6).
The same is true for the T_{ijk} second order coefficients (Initial coordinates normally defined with *OBJET*, $KOBJ = 6$).

OBJECT DEFINITION

Use *OBJET*, $KOBJ = 5$ for constraint type $IC = 0$ and $IC = 1$, and $KOBJ$ =6 for $IC = 2$.

For constraint type $IC = 3$, the object is normally defined with keyword *OBJET*. If $KOBJ \neq 1$, any of the 1 to $IMAX$ trajectories can be constrained. If $KOBJ = 2$, only the first seven trajectories can be constrained.

THE FITTING PROCEDURE [14]

The procedure is a direct sequential minimization of the quadratic sum of all errors (*i.e.*, differences between desired and actual values of 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.

GASCAT: Gas Scattering

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

MCDESINT: Monte Carlo Simulation of In-Flight Decay [15]

As soon as *MCDESINT* appears 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.
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 $D_1 = \frac{p_1}{q}\frac{1}{BORO}$ with $BORO$ = reference rigidity, see *OBJET*), and position Y_1, Z_1 in the **Zgoubi** frame. 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 decay length s_1 of particle 1 is related to its center of mass lifetime τ_1 by

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

The path length s up to the decay point is then calculated from a random number $0 < R_1 \leq 1$ by using the exponential decay formula

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

After decay, particle 2 will be ray-traced with assumed positive charge, while particle 3 is abandoned. 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

$$\begin{aligned} \theta^* &= 2\pi(R_2 - 0.5) \qquad (-\pi < \theta^* \leq \pi) \\ \phi &= 2\pi R_3 \qquad\qquad (0 < \phi \leq 2\pi) \end{aligned}$$

ϕ is a relativistic invariant, and θ in the laboratory frame (Fig. 7) is given by

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

where, β_2^* and momentum p_2 are given by

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

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 $D_2 = \frac{p_2}{q}\frac{1}{BORO}$ (where $BORO$ is the reference rigidity, see *OBJET*), while the starting position of M_2 is $Y_2 = Y_1$ and $Z_2 = Z_1$.

The decay simulation by **Zgoubi** obeys 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 $F(6, I) - \frac{XPAS}{2}$, 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].

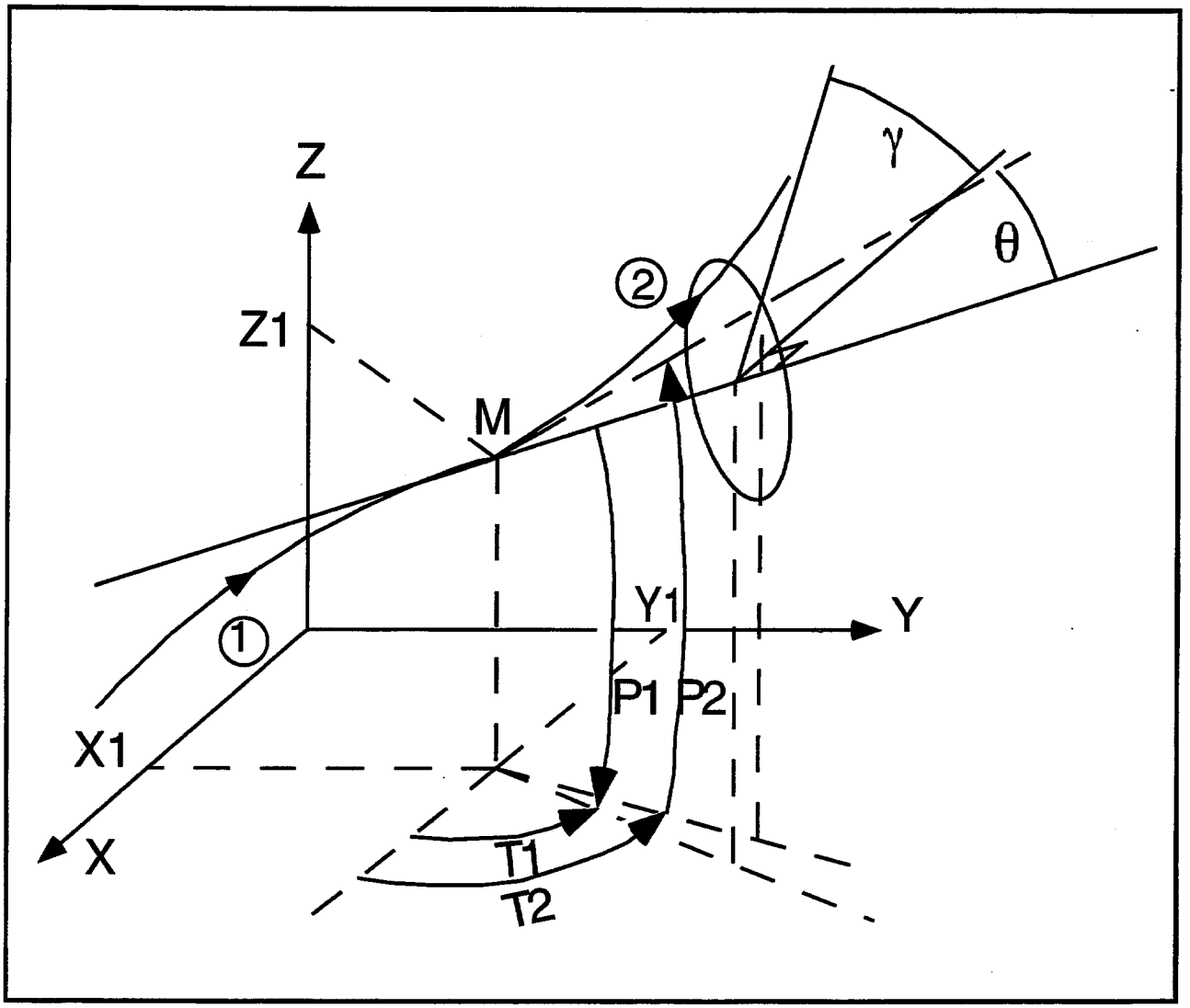


Figure 7: At position $M(X_1, Y_1, Z_1)$, particle 1 decays into 2 and 3; **Zgoubi** then calculates the trajectory of 2, while 3 is abandoned.
θ and ϕ are the scattering angles of particle 2 relative to the direction of the incoming particle 1; they transform to T_2 and P_2 in **Zgoubi** frame.

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, and 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. For the same purpose (e.g., use of *HISTO*), any particle of type 2 (resulting from decay of 1) will be tagged with an S standing for "secondary". When a particle decays, its coordinates D, Y, T, Z, P at the decay point are stored in $FDES(J, I)$, $J = 1, 5$.

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 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 spot of the primary particle from which it originated, then that primary particle is regenerated with its original coordinates at that spot. Then the secondary particle is abandoned while ray-tracing resumes in a regular way for the primary particle which is again susceptible of decay at the same time-of-flight. This procedure is made possible by prior storage of the coordinates of the primary particles (in array $FDES(J, I)$) each time a decay occurs.
Negative steps ($XPAS < 0$) in optical elements are not compatible with *MCDESINT*.

ORDRE: Higher Order Taylor Expansions in lenses

The position $\vec{R}$ and velocity $\vec{u}$ of a particle are obtained from Taylor expansions as described in eq. (2.2.4). By default, these expansions go to the fourth order derivative of $\vec{u}$,

$$\vec{R}_1 = \vec{R}_0 + \vec{u}_0 ds + ... + \vec{u}_0^{(4)} \frac{ds^5}{5!}$$
$$\vec{u}_1 = \vec{u}_0 + \vec{u}_0' \frac{ds^2}{2!} + \ldots + \vec{u}_0^{(4)} \frac{ds^4}{4!}$$

which corresponds to third order derivatives of $\vec{B}$, since (eq. (2.2.6))

$$\vec{u}^{(4)} = \vec{u}_1''' \times \vec{B} + 3\vec{u}_1'' \times \vec{B}' + 3\vec{u}_1' \times \vec{B}'' + \vec{u}_1 \times \vec{B}'''$$

and to the third order derivatives of $\vec{E}$ (eq. (2.2.10)). However the $\vec{B}'''$ or $\vec{E}'''$ term may be zero in second order type optical elements, for instance in a sharp edge quadrupole. Also, in several elements, not more than the first and second order derivatives of the fields are used.

The purpose of *ORDRE*, option $IO = 5$, is to allow expansions of $\vec{R}$ and $\vec{u}$ up to the term $\vec{u}^{(5)}$ for the following optical elements

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

The use of *ORDRE* with $IO = 4$ is equivalent to the default functioning.

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

PARTICUL: Particle Characteristics

PARTICUL allows the definition of several characteristics of the particles (mass, charge, gyromagnetic factor and life-time in the center of mass), that are needed in several procedures, as follows

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

The declaration of *PARTICUL* must **precede** these keywords.
Note that, in the case of electric or electro-magnetic optical elements, the mass and charge are needed in order to compute the particle velocity v, as involved in eq. 2.2.3.

REBELOTE: Jump To the Beginning of Zgoubi Input Data File

As soon as *REBELOTE* is encountered in the input data file, the code execution jumps back to the beginning of the data file to start a new run, and so on up to *NPASS* times. When the following random procedures are used: *MCOBJET, OBJETA, MCDESINT, SPNTRK* ($KSO = 5$), their random seeds are not reset, and therefore independent statistics will add up. *REBELOTE* is dedicated either to Monte Carlo calculations when more than 200 particles are to be tracked (due to $IMAX \leq 200$, see *MCOBJET*), or to the tracking in circular machines (e.g. Synchrotron accelerators). The option index K is then used to either generate new initial coordinates ($K = 0$ see section 5.6.7), when using *MCOBJET* or any other generator of random initial coordinates, or in order that the final coordinates at the last run be taken as the initial coordinates of the next ($K = 99$ — see section 5.6.4).

Monte Carlo simulations: normally $K = 0$. *NPASS* runs through the same structure will follow, resulting in the calculation of $(1 + NPASS) * IMAX$ trajectories.

Circular machines: normally $K = 99$. *NPASS* turns in the same structure will follow, resulting in the tracking of *IMAX* particles over $1 + NPASS$ turns (Note: for the simulation of accelerators and synchrotron motion, see *SCALING*).

Output prints over *NPASS* runs might result in a prohibitively big file. They may be inhibited by means of the option $KWRIT = 0$.

REBELOTE provides statistical calculations and related informations on particle decay *(MCDESINT)*, spin tracking *(SPNTRK)*, stopped particles (*CHAMBR, COLLIMA*).

RESET: Reset Counters and Flags

Piling up problems in **Zgoubi** input data file is allowed, with normally no particular precaution, except that each new problem must begin with a new object definition (with *MCOBJET*, *OBJET*, etc.). Nevertheless, when calling upon certain keywords, flags, counters or integrating procedures are involved. It may therefore be necessary to reset them. This is the purpose of *RESET* which normally appears right after the object definition and causes each problem to be treated as a new and independent one.

The keywords or procedures of concern and the effect of *RESET* are the following

CHAMBR	: *NOUT* = number of stopped particles = 0; *CHAMBR* option switched off
COLLIMA	: *NOUT* = number of stopped particles = 0
HISTO	: Histograms are emptied
INTEG	: *NRJ* = number of particles out of range = 0 (*INTEG* is the numerical integration subroutine; *NRJ is incremented when a particle goes out of a field map*)
MCDESINT	: Decay in flight option switched off
SCALING	: Scaling options disabled
SPNTRK	: Spin tracking option switched off

SCALING: Time Scaling of Power Supplies and R.F. Cavity

SCALING acts as a function generator dedicated to varying the field of optical elements, or the frequency in *CAVITE*. It is normally intended to be declared right after the object definition, and used in conjunction with *REBELOTE*, for the simulation of multiturn tracking with acceleration cycles.

SCALING acts on families of elements, a family being designated by a specific name, cataloged as such in the Program, and which coincides with the keyword of the corresponding element. For instance, declaring *MULTIPOL* as to be varied will result in the same timing law being applied to all *MULTIPOL*'s declared in the **Zgoubi** optical structure data file. Subsets can be selected by labeling keywords in the data file (section 5.6.3) and adding the corresponding label(s) in the *SCALING* declarations. The family name of concern, as well as the field versus timing scaling law of that family (or frequency versus timing in the case of *CAVITE*) are given as input data to the keyword *SCALING*. Up to 10 families can be declared as subject to a scaling law; a scaling law can be made of up to 10 successive timings; between two successive timings, the variation law is linear.

An example of data formatting is given in the following

SCALING		- Scaling
1 4		Active. 4 families of elements are concerned
QUADRUPO QF		- Quadrupoles labeled 'QF' (field at pole tip = B_o)
2		2 timings
18131.E-3	24176.E-3	B increases (linearly) from 18131E-3$*B_o$ to 24176E-3$*B_o$
1	6379	from turn 1 to turn 6379
MULTIPOL QD		- Multipoles labeled 'QD' (field of multipole component i at pole tip= B_io)
2		
18131.E-3	24176.E-3	B_i increases from 18131E-3$*B_io$ to 24176E-3$*B_io$
1	6379	from turn 1 to turn 6379
BEND		- All *BEND*'s (Bending magnets)
2		
18131.E-3	24176.E-3	Same scaling
1	6379	
CAVITE		- Accelerating cavity (reference frequency=f_{RFo})
2		
1 1.22	1.33352	The synchronous rigidity $(B\rho)_s$ increases from $(B\rho)_{s_o}$
1 1200	6379	to 1.22 $*(B\rho)_{s_o}$ and to 1.33352 $(B\rho)_{s_o}$
		from turn 1 to 1200 and 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

$$B(N) = SCALE(N)B_o$$
$$B_i(N) = SCALE(N)B_io$$

The cavity R.F. is calculated by

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

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,

$$\text{if } 1 \leq N \leq 1200 \; SCALE(N) = 1 + \frac{1.22 - 1}{1 + 1200 - 1}(N - 1)$$

$$\text{if } 1200 \leq N \leq 6379 \; SCALE(N) = 1.22 + \frac{1.33352 - 1.22}{1 + 6379 - 1200}(N - 1200)$$

and then,

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

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

Families amenable to scaling are, AIMANT, BEND, CAVITE, DECAPOLE, DIPOLE, DODECAPO, MULTIPOL, OCTUPOLE, POISSON, QUADISEX, QUADRUPO, SEXQUAD, SEXTUPOL, SOLENOID, TOSCA, UNDULATOR.

SPNTRK: Spin Tracking

The keyword *SPNTRK* permits switching on the spin tracking option. It also permits the attribution of an initial spin component 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 the mass and gyromagnetic factor.

KSO = 1 (respectively 2, 3): the *IMAX* particles of the beam 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 = 5: random generation of *IMAX* initial spin conditions as described in Fig. 8. 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. 9.

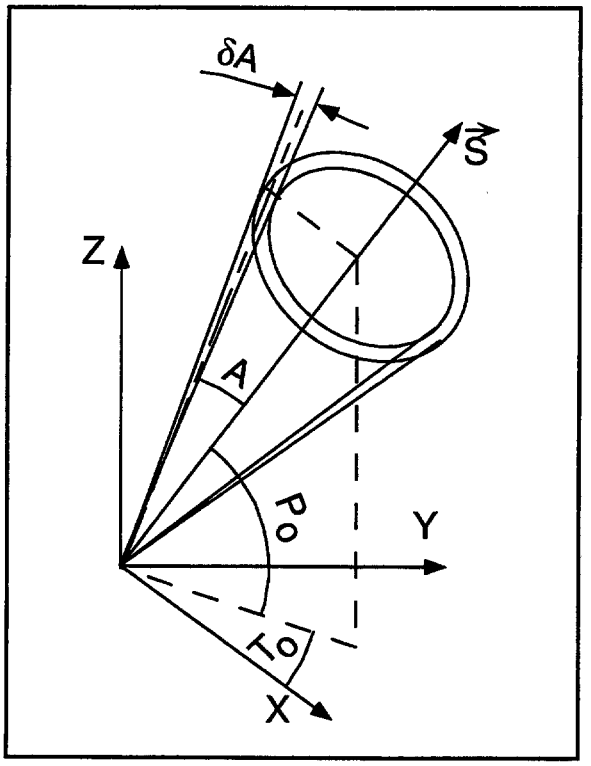


Figure 8: Spin distribution as obtained with option $KSO = 5$.
The spins are distributed within an annular strip δA (standard deviation) at an angle A with respect to the axis of mean polarization (S) defined by T_0 and P_0.

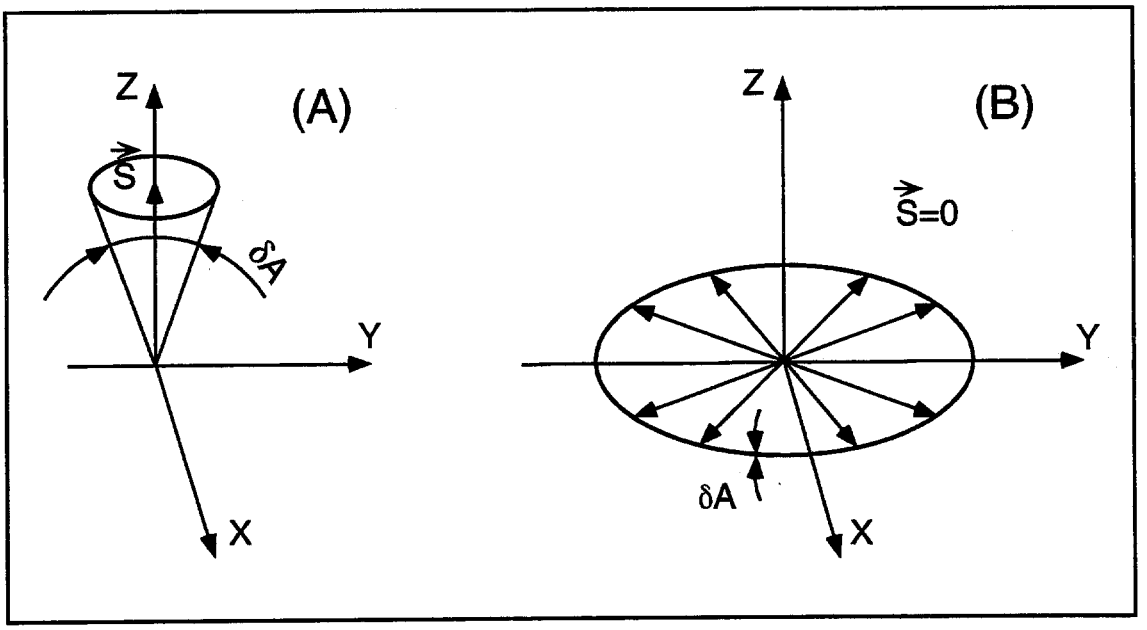


Figure 9: Examples of the use of KSO =5.
A: Gaussian distribution around a mean vertical polarization axis, obtained with T_0 = arbitrary, $P_0 = \pi/2$, $A = 0$ and $\delta A \neq 0$.
B: Isotropic distribution in the median plane, obtained with $P_0 = \pm\pi/2$, $A = \pi/2$, and $\delta A = 0$.

SYNRAD: Synchrotron Radiation

The keyword *SYNRAD* enables (or disables) the calculation of synchrotron radiation (SR) electric field and brightness. It must be preceded by *PARTICUL* for the definition of the charge.

SYNRAD is supposed to appear a first time at the location where SR calculations should start, 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. 4.1.1)
$(btx, bty, btz) = \vec{\beta} = \frac{1}{c} \times$ particle velocity
$(gx, gy, gz) = \frac{d\vec{\beta}}{dt}$ = particle acceleration (eq. 4.1.3)
$\Delta\tau$ = observer time increment (eq. 4.1.2)
$t' = \tau - r(t')/c$ = retarded (particle) time
$(rtx, rty, rtz) : \vec{R}(t)$, particle to observer vector (eq. 4.1.4)
(x, y, z) = particle coordinates
ds = step size in the magnet (fig. 2)
NS = step number
I = particle number
$LET(I)$ = tagging letter
$IEX(I)$ = stop flag (see section 5.6.8)

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 brightness $\int_{\nu_1}^{\nu_2} \partial^3 P/\partial\phi\,\partial\psi\,\partial\nu$ (after eq. 4.2.2) following Fourier transform of the electric field (eq. 4.2.2). The spectral range of interest and frequency sampling : ν_1, ν_2, N, is specified next to *KSR*.

Note that $KSR = 0$ followed by a dummy line of data allows temporary inhibition of SR procedures.

5.4 Optical Elements and related numerical procedures

AIMANT: Generation of a Dipole Magnet 2-D Map

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*. The extent of the map is defined by the following parameters, as shown in Figs. 10A and 10B.

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 EFB's.
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

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

where N, B and G are respectively the first, second and third order field indices and $\mathcal{F}$ is the fringe field coefficient, while the X and Y components of the field are assumed to be zero on the map mesh.

Calculation of the Fringe Field Coefficient

With each EFB a realistic extent of the fringe field, λ, is associated (Figs. 10A and 10B), 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 $\mathcal{F} = 1$ 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. 11) given by

$$\begin{aligned} F &= \frac{1}{2}\frac{(\lambda - s)^2}{\lambda^2 - \xi^2} \quad \text{if } \xi \leq s \leq \lambda \\ F &= 1 - \frac{1}{2}\frac{(\lambda - s)^2}{\lambda^2 - \xi^2} \quad \text{if } -\lambda \leq s \leq -\xi \end{aligned} \tag{5.4.2}$$

where s is the distance to the EFB, and

$$\begin{aligned} F &= \frac{1}{2} + \frac{s}{\lambda + \xi} \quad \text{if } 0 \leq s \leq \xi \\ F &= \frac{1}{2} - \frac{s}{\lambda + \xi} \quad \text{if } -\xi \leq s \leq 0 \end{aligned} \tag{5.4.3}$$

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 more accuracy it is better to use the next option.

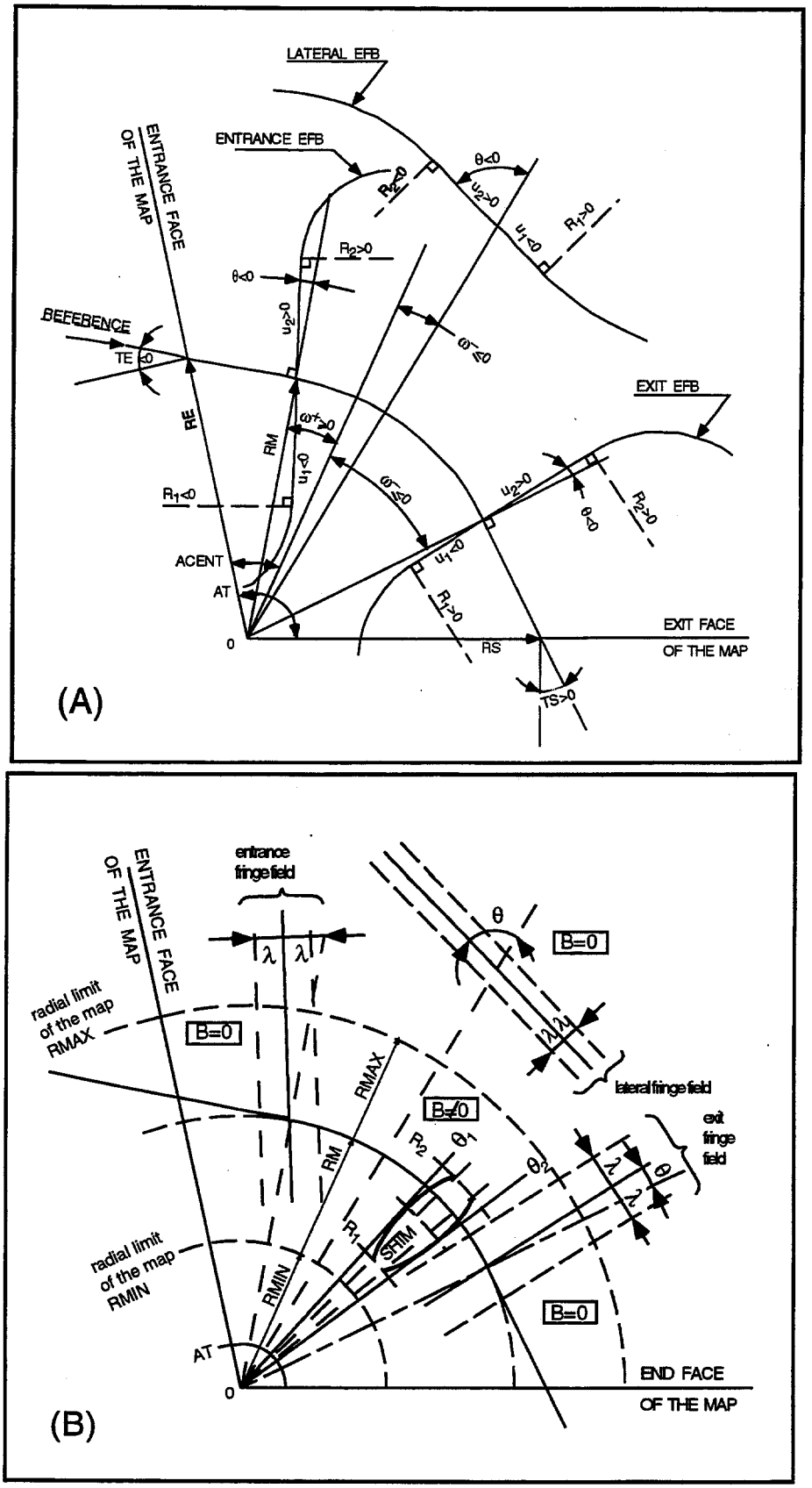


Figure 10: A: Parameters used to define the field map and geometric boundaries.
B: Parameters used to define the field map and fringe fields.

If $\xi = -1$, F is an exponential type fringe field (Fig. 11) given by [16]

$$F = \frac{1}{1 + \exp P(s)} \tag{5.4.4}$$

where s is the distance to the EFB, and

$$P(s) = C_0 + C_1\left(\frac{s}{\lambda}\right) + C_2\left(\frac{s}{\lambda}\right)^2 + C_3\left(\frac{s}{\lambda}\right)^3 + C_4\left(\frac{s}{\lambda}\right)^4 + C_5\left(\frac{s}{\lambda}\right)^5 \tag{5.4.5}$$

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 (this restriction no longer holds in the improved version *DIPOLE*).
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 eqs. above. At any node of the map mesh, the resulting value of the fringe field coefficient (eq. 5.4.1) is (Fig. 12)

$$\mathcal{F} = F_E * F_S * F_L$$

($F_L = 1$ 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

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

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. 10B). *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 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

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

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

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

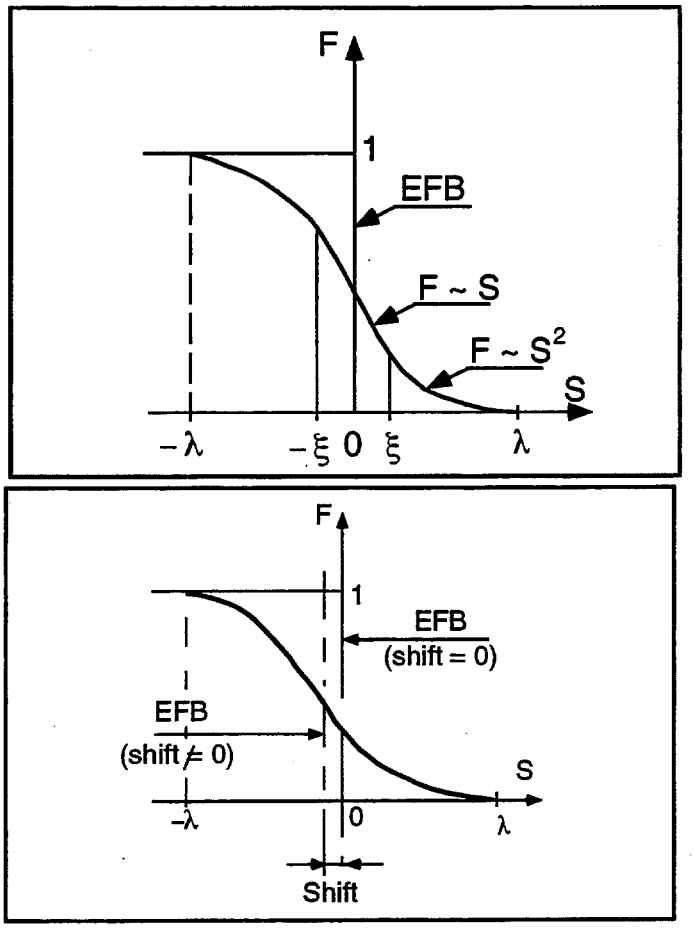


Figure 11: Second order type fringe field (leftt plot) and exponential type fringe field (right plot)..

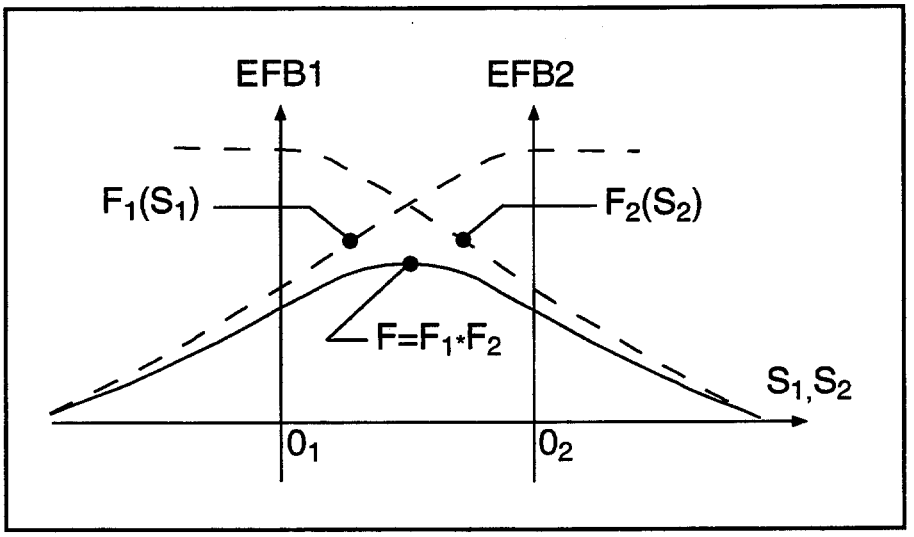


Figure 12: Effective value of $\mathcal{F}$ for overlapping fringe fields F_1 and F_2 centered at O_1 and O_2.

If NBS ≥ 1, then *NBS* shims are introduced at positions $\frac{R_1+R_2}{2}$, $\frac{\theta_1+\theta_2}{2}$ (Fig. 13) [17]
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. 13, $\rho = \frac{R_1+R_2}{2}$ 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 2.4.2). The transformation from polar to Cartesian coordinates is performed following eqs. (2.4.9 or 2.4.10). Extrapolation off median phase is then performed by means of Taylor expansions following the procedure described in section 2.3.2.

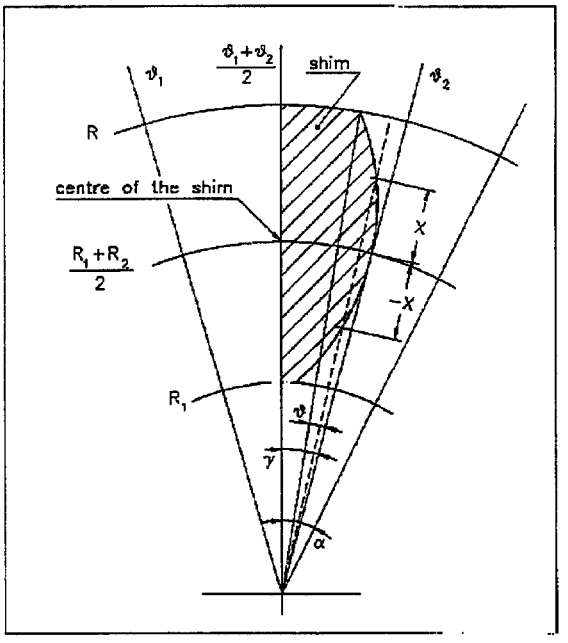


Figure 13: A second order profile shim. The shim is centered at $\frac{(R_1+R_2)}{2}$ and $\frac{(\theta_1+\theta_2)}{2}$.

AUTOREF: Automatic transformation to a new reference frame

AUTOREF positions the new reference frame following 3 options:

If I = 1, *AUTOREF* is equivalent to

$$CHANGREF[XCE = 0, YCE = Y(1), ALE = T(1)]$$

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 I = 2, it is equivalent to

$$CHANGREF[XW, YW, T(1)]$$

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 $T(1)$ is set to zero.

If I = 3, it is equivalent to

$$CHANGREF[XW, YW, T(I1)]$$

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 $T(1)$ is set to zero.

BEND: Bending magnet

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

The first input data are the magnet length XL and the field B_o, such that in absence of fringe field the deviation θ verifies $XL = 2\frac{BORO}{B_o}\sin(\theta/2)$ ($BORO$ = reference rigidity, defined in *OBJET*).

Then follows the description of the entrance and exit EFB's and fringe fields. The model is the same as for *DIPOLE*. The wedge angles W_E (entrance) and W_S (exit) are defined with respect to the sector magnet, with the signs described in Fig. 14. Within a distance $\pm X_E(\pm X_S)$ on both sides of the entrance (exit) EFB, the fringe field model is used; 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 set 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-)alignement is assured by *KPOS*, with special features allowing some degrees of automatism useful for periodic structures (section 5.6.5).

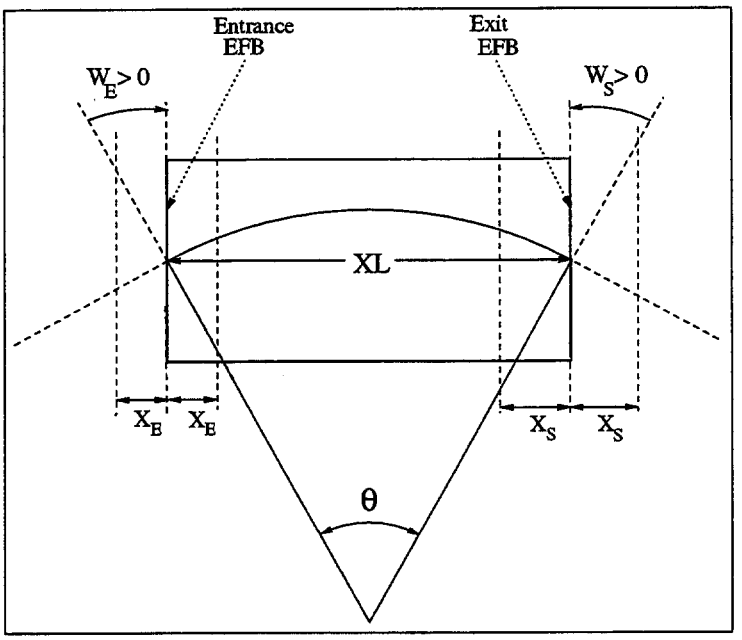


Figure 14: Geometry and parameters in *BEND*. XL = length, B_o = field, θ = deviation. The particular case of parallel face bend is illustrated here, obtained by setting $W_E = W_S = \theta/2$.

BREVOL: 1-D Uniform Mesh Magnetic Field Map

BREVOL 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), 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 $BX(I)$ the values of the X component of the field. BX is normalized with *BNORM* prior to ray-tracing. For binary files, *FNAME* must begin with 'B_ ', 'BINARY' will then be set to '.TRUE.'.

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 fit followed by second order off-axis Taylor series extrapolation (see sections 2.3.1, 2.4.1).

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 Mesh Magnetic Field Map With Mid-Plane Symmetry

CARTEMES was originally dedicated to the reading and processing of the measured median plane field maps of the QDD spectrometer SPES2 at Saclay. However, it can be used for the reading of any other 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 nodes of the uniform mesh, and $X(I)$, $Y(J)$ their coordinates. *FNAME* is the file containing the field data. For binary files, *FNAME* must begin with 'B_ ', 'BINARY' will then be set to '.TRUE.'.

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

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

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 2.4.2).

Entrance and/or exit integration boundaries can be defined with the flag ID, as follows (Fig. 15).

If ID = 1: the integration in the field is stopped on a boundary with equation $A'X + B'Y + C' = 0$, and then the trajectories are extrapolated linearly onto the exit end 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 end, 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 stops on the last $(ID - 1)$ exit boundary. No extrapolation onto the map exit end is performed in this case.

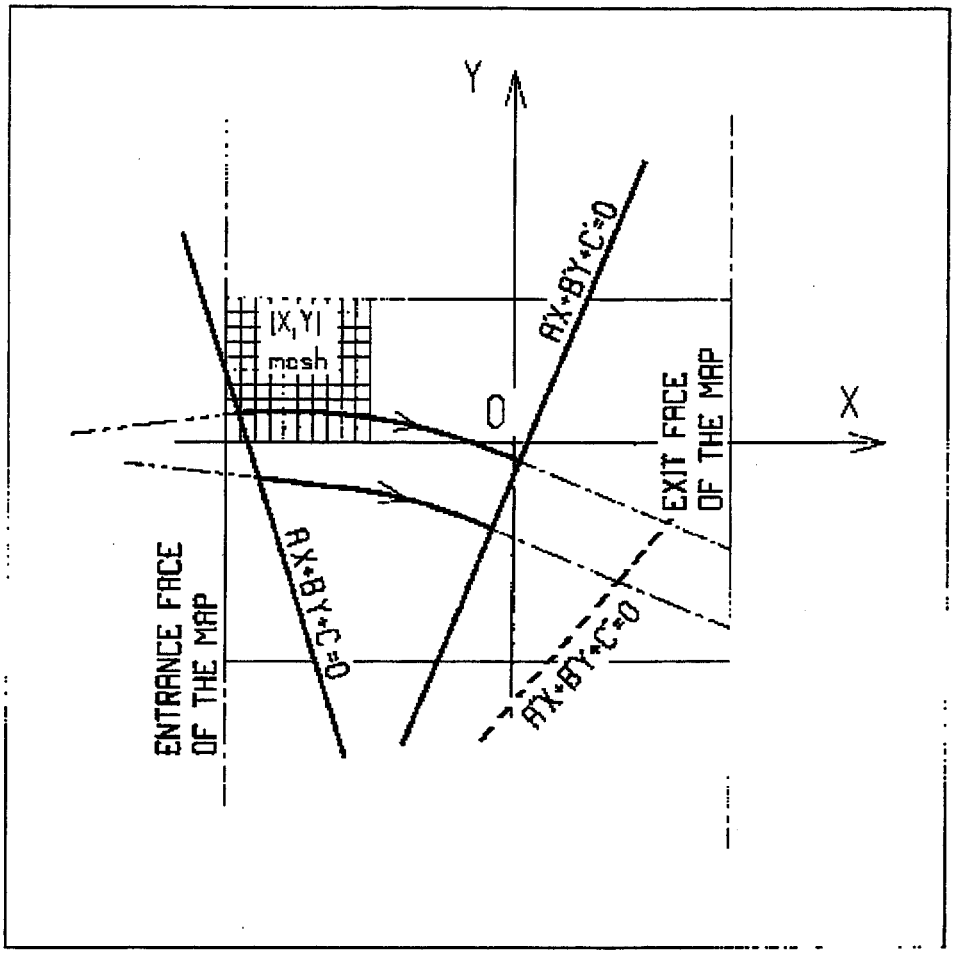


Figure 15: OXY is the coordinate system of the mesh. Integration boundaries may be defined, using $ID \neq 0$: particle coordinates are extrapolated linearly from the entrance face of the map, onto the boundary $A'X + B'Y + C' = 0$; after ray-tracing inside the map and stopping on the boundary $AX + BY + C = 0$, coordinates are extrapolated linearly onto the exit face of the map if $ID = 2$, or stopped on the last $(ID - 1)$ boundary if $ID > 2$.

CAVITE: Accelerating Cavity

CAVITE provides an analytical simulation of a (zero length) accelerating cavity; it can be used in conjunction with keywords *REBELOTE* and *SCALING* for the simulation of multiturn tracking with synchrotron acceleration (see section 5.6.7). It must be preceded by *PARTICUL* for the definition of mass M and charge q.

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

If IOPT = 1: *CAVITE* simulates the R.F. cavity of a synchrotron accelerator. Normally the keyword *CAVITE* appears **at the end** of the optical structure (the periodic motion over $IT = 1$, $NPASS + 1$ turns is simulated by means of the keyword *REBELOTE*, option K = 99 and the R.F. and magnetic fields timings by means of *SCALING* — see section 5.6.7). The synchrotron motion of any of the *IMAX* particles of a beam is obtained by solving 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 = R.F. phase; $\phi_2 - \phi_1$ = variation of ϕ between two traversals
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 the bottom NOTE)
ℓ = orbit length of the particle between two traversals
$\beta_s c$ = velocity of the (virtual) synchronous particle
βc = velocity of the particle
$\hat{V}$ = peak R.F. voltage
q = particle electric charge.

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

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

where

h = harmonic number of the R.F
M = mass of the particle
c = velocity of light.

The current rigidity $(B\rho)_s$ of the synchronous particle is obtained from the timing law specified by means of *SCALING* following $(B\rho)_s = BORO \cdot SCALE(TIMING)$ (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* 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

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

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

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

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

Calculation of the coordinates

Let $p_I = \left[p_{XI}^2 + p_{YI}^2 + p_{ZI}^2\right]^{1/2}$ 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 IOPT = 2 : the same simulation of a synchrotron R.F. cavity, as for **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 given in the input data. From this, the calculation of the law $(B\rho)_s$ and the R.F. frequency f_{RF} follows, according to the formulae given in $IOPT = 1$.

If IOPT = 3: 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.

NOTE: Calculation of the closed orbit.
Due to the fringe fields, the horizontal closed orbit may not coincide with the ideal axis of the optical elements. One way to calculate it at the beginning of the structure (*i.e.* where the initial particle coordinates have to be defined) is to ray-trace a single particle over a significantly high number of turns, starting with the initial condition ($Y_0 = T_0 = Z_0 = P_0 = 0$), and so as to obtain a statistically well-defined phase-space ellipse. The initial conditions of the closed orbit then correspond to the coordinates Y_c and T_c of the center of this ellipse. Next, ray-tracing over one turn a particle starting with the initial condition (Y_c, T_c, $Z_0 = P_0 = 0$) will provide the length $\mathcal{L}$ (namely, the $F(6,1)$ coordinate) of the closed orbit.

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 verify

$$(Y - YC)^2 \geq YL^2 \text{ or } (Z - ZC)^2 \geq ZL^2 \quad \text{if} \quad IFORM = 1$$
$$\frac{(Y - YC)^2}{YL^2} + \frac{(Z - ZC)^2}{ZL^2} \geq 1 \quad \text{if} \quad IFORM = 2$$

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 magnets 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 $XL + X_S$, 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 polar coordinate magnets, Y stands for the radial coordinate (e.g. with *DIPOLE*, see Figs. 3C and 10). Therefore, centering *CHAMBR* at $YC = RM$ simulates a chamber curved with radius RM, and having a radial acceptance $RM \pm YL$. The testing is done in *ESL* (*DRIFT*) at the beginning and 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 5.6.8) is set to the value -4, and its actual path length is stored in the array *SORT* for eventual further statistical purposes.

CHANGREF: Transformation to a New Reference Frame

CHANGREF transports the particles to a new reference frame. It can be used anywhere in a structure. The new coordinates of the particles Y_2, T_2, Z_2 and P_2 and the path length S_2 are deduced from the old ones Y_1, T_1, Z_1, P_1 and S_1 by

$$
\begin{aligned}
T_2 &= T_1 - ALE \\
Y_2 &= \frac{(Y_1 - YCE)\cos T_1 + XCE \sin T_1}{\cos T_2} \\
DL^2 &= (XCE - Y_2 \sin ALE)^2 + (YCE - Y_1 + Y_2 \cos ALE)^2 \\
Z_2 &= Z_1 + DL \mathrm{tg} P_1 \\
S_2 &= S_1 + \frac{DL}{\cos P_1} \\
P_2 &= P_1
\end{aligned}
$$

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 $XCE - Y_2 \sin(ALE)$. This keyword may for instance be used for positioning optical elements, or for setting a reference frame at the entrance or exit of field maps. Effects of *CHANGREF* on spin tracking, particle decay and gas-scattering are taken into account (but not on synchrotron radiation).

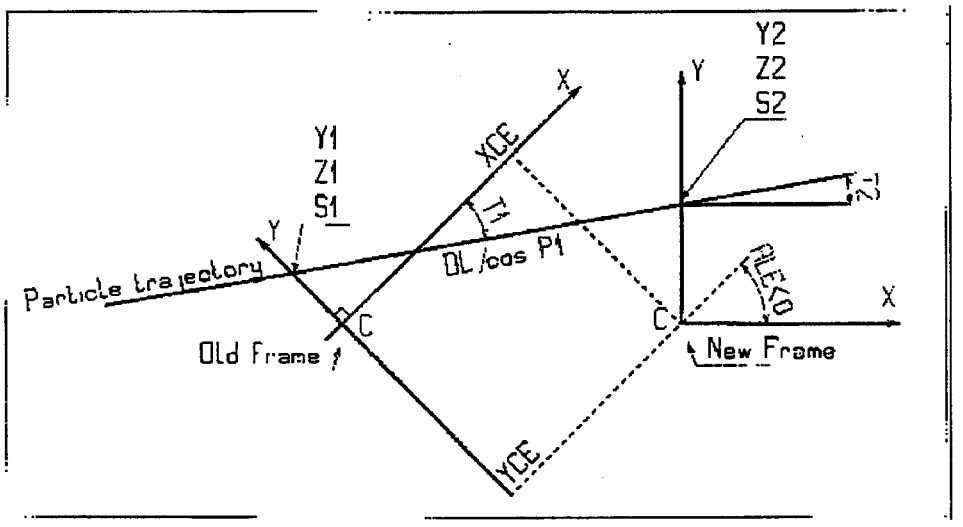


Figure 16: Scheme of the *CHANGREF* procedure.

CIBLE or TARGET: Generate a Secondary Beam From 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. 17.

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

$$CHANGREF(XCE = YCE = 0, \quad ALE = \beta)$$

followed by

$$CHANGREF(XCE = YCE = 0, \quad ALE = \theta - \beta).$$

Particle 4 is abandoned, 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

$$M_1c^2 + W_2 = W_3 + W_4$$
$$p_4^2 = p_2^2 + p_3^2 - 2p_2p_3\cos(\theta - T)$$

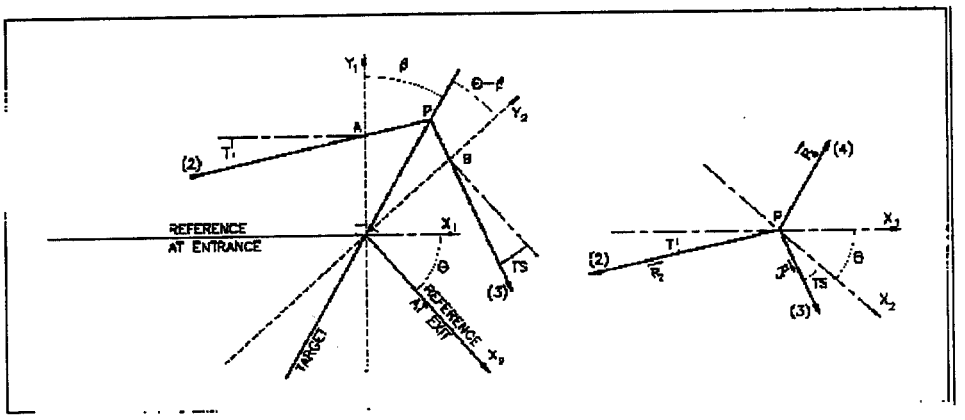


Figure 17: Scheme of the principles of *CIBLE (TARGET)*
A, T = position, angle of incoming particle 2 in the entrance reference frame
P = position of the interaction
B, T = position, angle of the secondary particle in the exit reference frame
θ = angle between entrance and exit frames
β = tilt angle 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 transverse limits of the aperture, which 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 verify

$$(Y - YC)^2 \geq YL^2 \text{ or } (Z - ZC)^2 \geq ZL^2 \quad \text{if} \quad IFORM = 1$$
$$\frac{(Y - YC)^2}{YL^2} + \frac{(Z - ZC)^2}{ZL^2} \geq 1 \quad \text{if} \quad IFORM = 2$$

When a particle is stopped, its index *IEX* (see *OBJET* and section 5.6.8) is set to the value -4, and its actual path length is stored in the array *SORT* for eventual further statistical purposes (e.g. with *HISTO*).

DECAPOLE: Decapole Magnet (Fig. 18)

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 approximated to the 5th order in Y and Z

$$V(X,Y,Z) = V(X,Y,Z) = G\left(Y^4Z - 2Y^2Z^3 + \frac{Z^5}{5}\right)$$

$$\text{with } G_0 = \frac{B_0}{R_0^4}$$

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

$$B_X = 0$$
$$B_Y = 4G_0(Y^2 - Z^2)YZ$$
$$B_Z = G_0(Y^4 - 6Y^2Z^2 + Z^4)$$

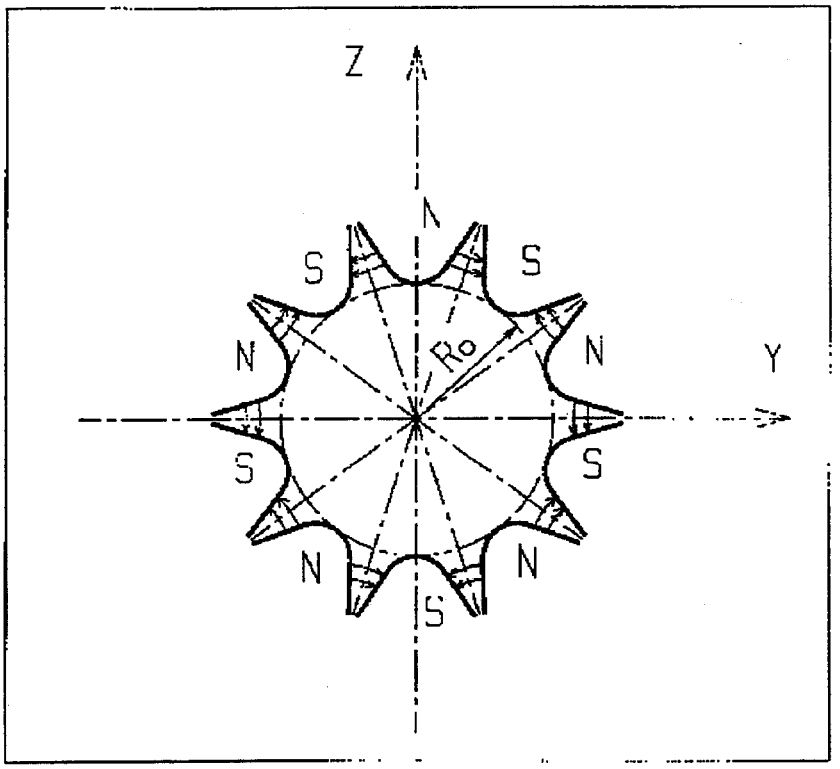


Figure 18: Decapole magnet

DIPOLE: Generation of a Dipole Magnet 2-D Map

DIPOLE is a recent, simpler and improved version of *AIMANT*.

The keyword *DIPOLE* 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. 10A and 10B.

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 EFB's 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

$$B = \mathcal{F} * B_0 * \left(1 + N * \left(\frac{R - RM}{RM}\right) + B * \left(\frac{R - RM}{RM}\right)^2 + G * \left(\frac{R - RM}{RM}\right)^3\right) \tag{5.4.6}$$

where N, B and G are respectively the first, second and third order field indices and $\mathcal{F}$ is the fringe field coefficient, while the X and Y components of the field are assumed to be zero on the mesh plane.

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. 11) given by [16]

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

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. 5.4.6) is

$$\mathcal{F} = F_E * F_S * F_L$$

($F_L = 1$ 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. 10B). *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 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 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 NBS ≥ 1, then *NBS* shims are introduced at positions $\frac{R_1 + R_2}{2}$, $\frac{\theta_1 + \theta_2}{2}$ (Fig. 13) [17]
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. 11, $\rho = \frac{R_1 + R_2}{2}$ 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 2.4.2). The transformation from polar to Cartesian coordinates is performed following eqs (2.4.9 or 2.4.10). Extrapolation off median plane is then performed by means of Taylor expansions, following the procedure described in section 2.3.2.

DODECAPO: Dodecapole Magnet (Fig. 19)

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

$$V(X,Y,Z) = V(X,Y,Z) = G\left(Y^4 - \frac{10}{3}Y^2Z^2 + Z^4\right)YZ$$

$$\text{with } G_0 = \frac{B_0}{R_0^5}$$

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

$$B_X = 0$$
$$B_Y = G_0(5Y^4 - 10Y^2Z^2 + Z^4)Z$$
$$B_Z = G_0(Y^4 - 10Y^2Z^2 + 5Z^4)Y$$

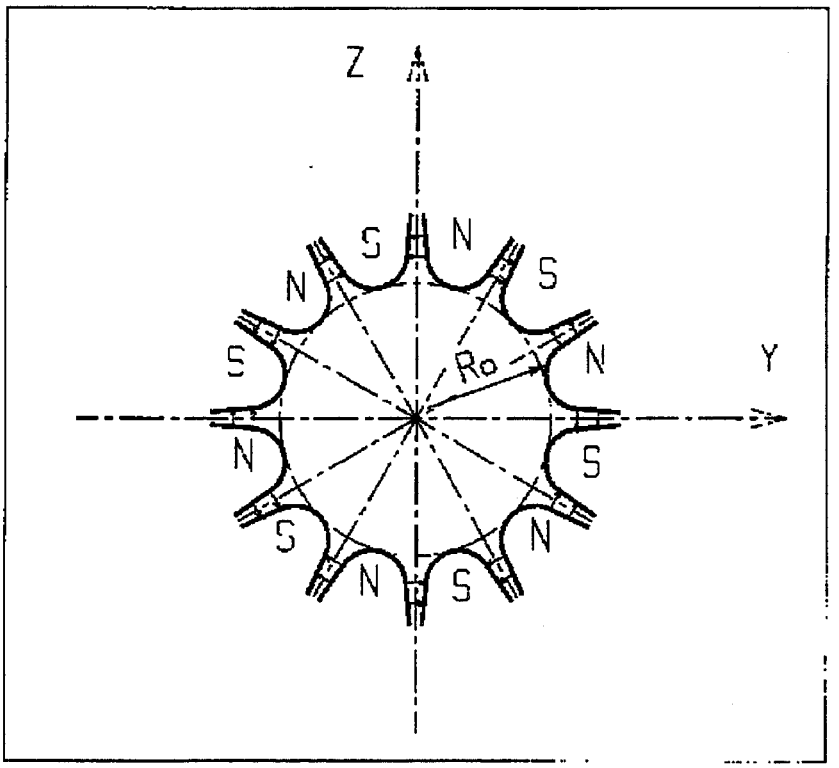


Figure 19: Dodecapole magnet

DRIFT or ESL: Field-Free Drift Space

DRIFT(or *ESL*) allows the introduction of a drift space with length XL with positive or negative sign, anywhere in a structure. The associated equations of motion are (Fig. 20)

$$Y_2 = Y_1 + XL * \mathrm{tg}T$$
$$Z_2 = Z_1 + \frac{XL}{\cos T} \mathrm{tg}P$$
$$SAR_2 = SAR_1 + \frac{XL}{\cos T * \cos P}$$

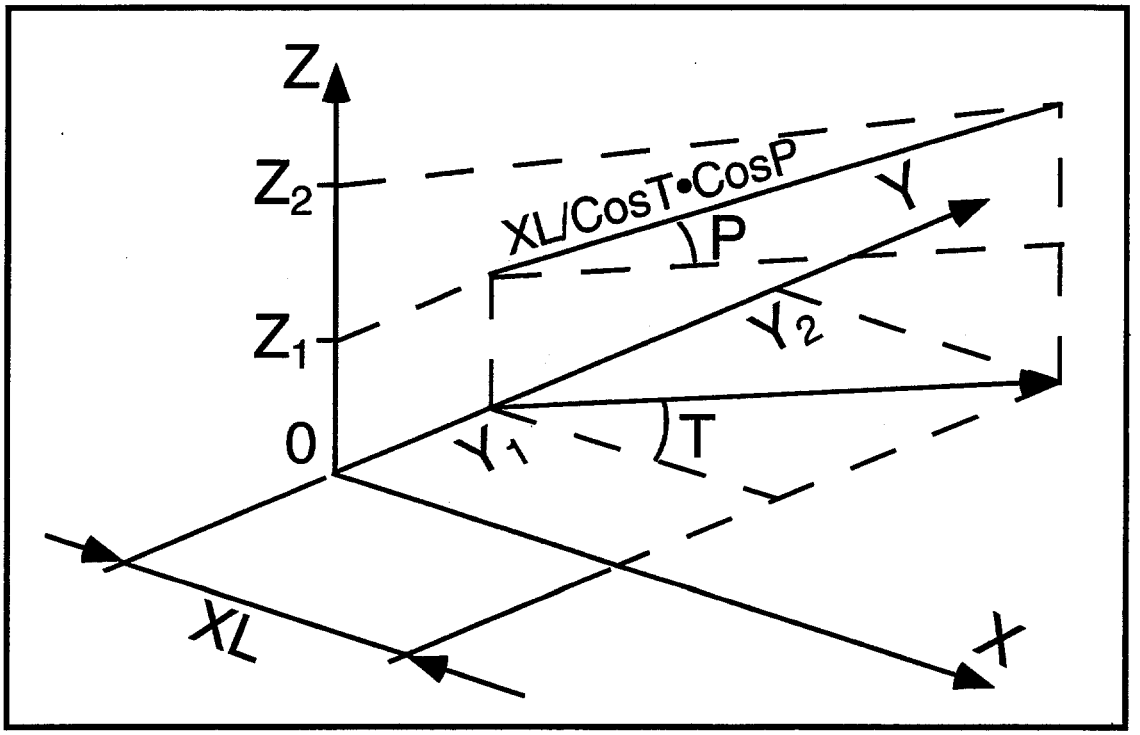


Figure 20: Transfer of particles in a drift space.

EBMULT: Electromagnetic Multipole

EBMULT simulates an electromagnetic multipole, by addition of electric ($\vec{E}$) and magnetic ($\vec{B}$) multipole components (dipole to dodecapole). $\vec{E}$ and its derivatives $\frac{\partial^{i+j+k}\vec{E}}{\partial X^i \partial Y^j \partial Z^k}$ are derived from the general expression of the multipole scalar potential (eq. 2.3.5), followed by a $\frac{\pi}{2n}$ rotation (n = pole order), as described in section 2.5.3 (see also *ELMULT*). $\vec{B}$ and its derivatives are derived from the same general potential, as described in section 2.3.5 (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 rotated independently of the others.
Use *PARTICUL* for the definition of the particle mass and charge.

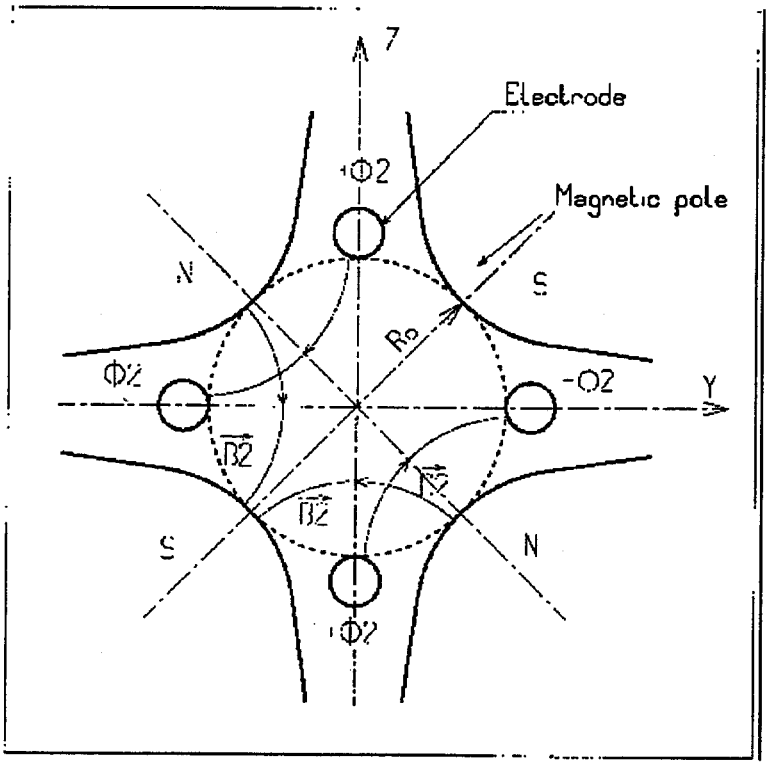


Figure 21: An example of $\vec{E}$, $\vec{B}$ multipole: the achromatic quadrupole [18] (known for its allowing null second order chromatic aberration).

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. 22). X-axis cylindrical symmetry is assumed. The model of electric potential along the axis is [20]

$$V(X) = \frac{V_2 - V_1}{2} \operatorname{th} \frac{\omega x}{R_0} \left[+ \frac{V_1 + V_2}{2} \right] \qquad \text{if } D = 0$$

$$V(X) = \frac{V_2 - V_1}{2} \frac{1}{2\omega D} \ell n \frac{\operatorname{ch} \omega \dfrac{x + D}{R_0}}{\operatorname{ch} \omega \dfrac{x - D}{R_0}} \left[+ \frac{V_1 + V_2}{2} \right] \qquad \text{if } D \neq 0$$

(x = distance from half-way between the electrodes; $\omega = 1.318$; 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 2.5.2 (note that they don't depend on the constant term $\left[\dfrac{V_1 + V_2}{2} \right]$ which disappears when differentiating).
Use *PARTICUL* for the definition of the particle mass and charge.

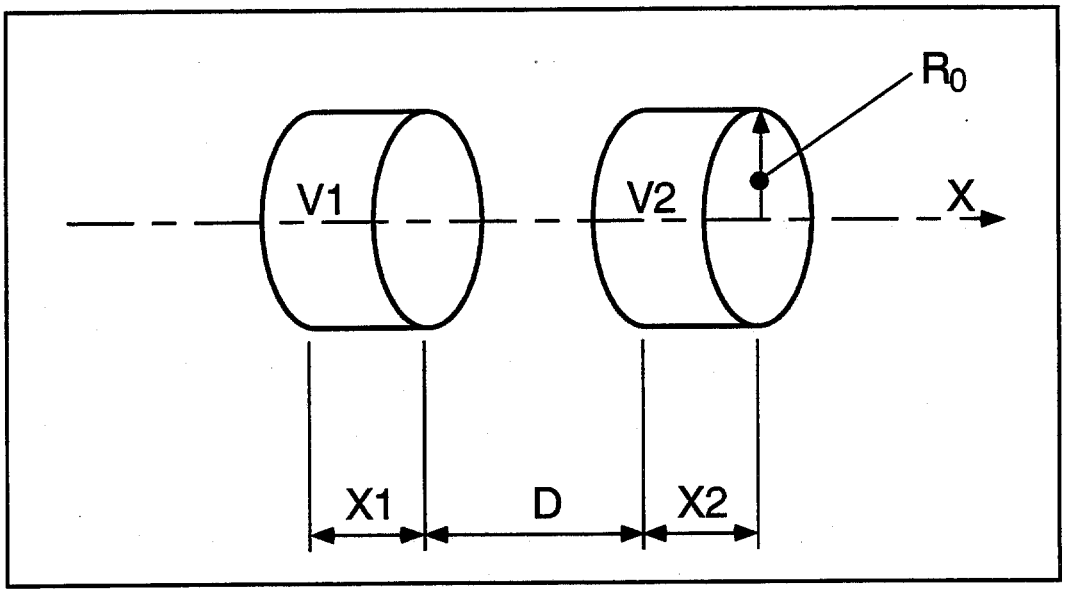


Figure 22: Two-electrode cylindrical electric lens.

ELMULT: Electric Multipole

The simulation of an electric multipolar field $\vec{M}_E$ proceeds by addition of the dipolar ($\vec{E}1$) to dodecapolar ($\vec{E}6$) components, and of their derivatives up to fourth order, following

$$\begin{aligned}
\vec{M}_E &= \vec{E}1 + \vec{E}2 + \vec{E}3 + \vec{E}4 + \vec{E}5 + \vec{E}6 \\
\frac{\partial \vec{M}_E}{\partial X} &= \frac{\partial \vec{E}1}{\partial X} + \frac{\partial \vec{E}2}{\partial X} + \frac{\partial \vec{E}3}{\partial X} + \frac{\partial \vec{E}4}{\partial X} + \frac{\partial \vec{E}5}{\partial X} + \frac{\partial \vec{E}6}{\partial X} \\
\frac{\partial^2 M_E}{\partial X \partial Z} &= \frac{\partial^2 \vec{E}1}{\partial X \partial Z} + \frac{\partial^2 \vec{E}2}{\partial X \partial Z} + \frac{\partial^2 \vec{E}3}{\partial X \partial Z} + \frac{\partial^2 \vec{E}4}{\partial X \partial Z} + \frac{\partial^2 \vec{E}5}{\partial X \partial Z} + \frac{\partial^2 \vec{E}6}{\partial X \partial Z} \\
&\text{etc.}
\end{aligned}$$

The independent components $\vec{E}1$ to $\vec{E}6$ and their derivatives up to the second order are calculated by differentiating the general multipole potential (2.3.5)

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

where $G(X)$ is the longitudinal gradient (see *QUADRUPO*) **but** including a $\frac{\pi}{2n}$ rotation about the X-axis, so that the so defined right electric multipole of order n, and of strength [18, 19]

$$K_n = \frac{1}{2} \frac{\gamma}{\gamma^2 - 1} \frac{\phi_n}{R_0^n}$$

(ϕ_n = potential at the electrode, R_0 = radius at pole tip, γ = relativistic Lorentz factor of the particle) has the same focusing effect than 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*).
Such $\frac{\pi}{2n}$ rotation of the multipole components is obtained following the procedure described in section 2.5.3.

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, octupolar, decapolar, dodecapolar) component is given by a coefficient $E2$ ($E3$, $E4$, $E5$, $E6$) at entrance, and $S2$ ($S3$, $S4$, $S5$, $S6$) at exit, such that the extent is $\lambda_E * E2$ ($\lambda_E * E3$, $\lambda_E * E4$, $\lambda_E * E5$, $\lambda_E * E6$) at entrance and $\lambda_S * S2$ ($\lambda_S * S3$, $\lambda_S * S4$, $\lambda_S * S5$, $\lambda_S * S6$) 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 $E_i = 0$ ($S_i = 0$) , the entrance (exit) fringe field for multipole component i is considered as a sharp edge field.

Overlapping of fringe fields inside the element is treated separately for each component, in the way described in *QUADRUPO*.
Moreover, any multipole component $\vec{E}_i$ can be rotated independently by an angle R_{X_i} around the longitudinal X-axis, for the simulation of positioning defects, as well as skewed lenses.
Use *PARTICUL* for the definition of the particle mass and charge.

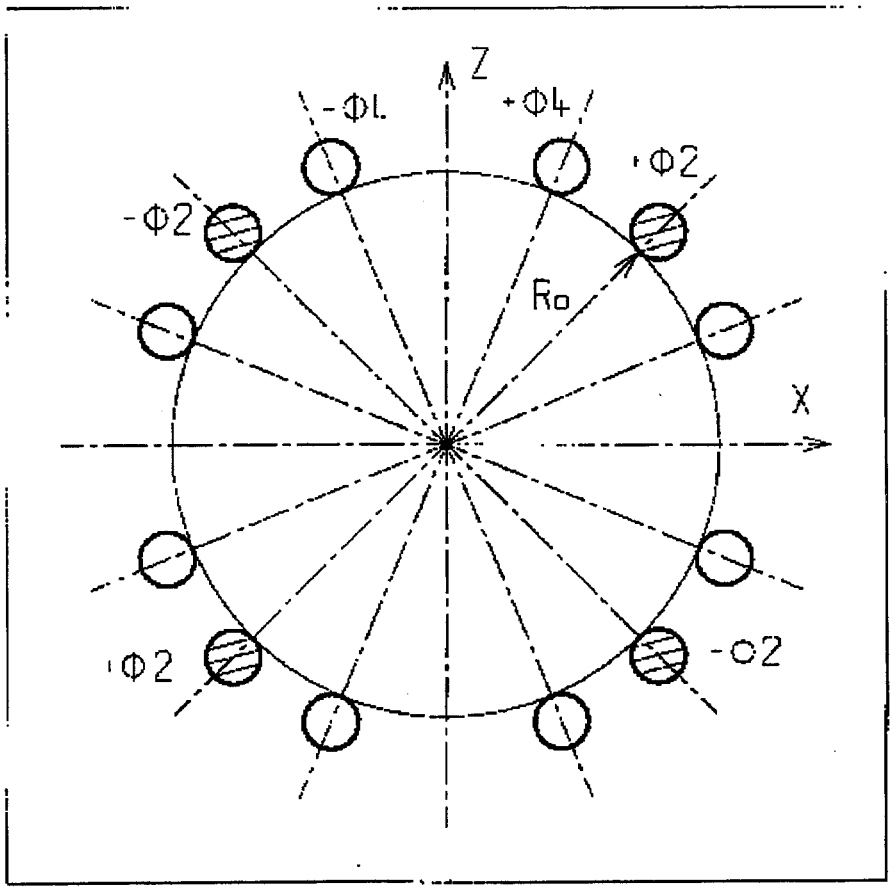


Figure 23: An electric multipole combining quadrupole ($\vec{E}2$) and octupole ($\vec{E}4$) component [19]
($\vec{E}1 = \vec{E}3 = \vec{E}5 = \vec{E}6 = 0$).

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 $EX(I)$ the values of the X component of the field. EX is normalized with *ENORM* prior to ray-tracing.

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 fit followed by second order off-axis Taylor series extrapolation (see sections 2.5.1 and 2.6).

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* for the definition of the particle mass and charge.

MAP2D: 2-D Cartesian Uniform Mesh Magnetic Field Map - no symmetry [21]

MAP2D reads a 2-D field map that provides the three components B_X, B_Y, B_Z of the magnetic field at the nodes of a 2-D Cartesian uniform mesh. No particular symmetry is assumed, which allows the treatment of any-type of field (e.g. dipole field out of median plane, solenoidal field). These data should be filed with a format that fits the following *FORTRAN* reading sequence involved in **Zgoubi**

```
      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), Z0, X(I), BY(I,J), BZ(I,J), BX(I,J)
             ELSE
                READ(NL,100) Y(J), Z0, X(I), BY(I,J), BZ(I,J), BX(I,J)
100             FORMAT (1X, 6E11.2)
             ENDIF
      1       CONTINUE
```

where IX (JY) is the number of longitudinal (transverse) nodes of the 2-D uniform mesh, Z_0 is the Z-elevation of the map. For binary files, FNAME must begin with 'B_ ', 'BINARY' will then be set to '.TRUE.'. The field $\vec{B} = (B_X, B_Y, B_Z)$ is next normalized with BNORM, prior to ray-tracing.

At each step of the trajectory of a particle, the field and its derivatives are calculated by a polynomial interpolation followed by a Z extrapolation (see sections 2.3.3, 2.4.3). Entrance and/or exit integration boundaries may be defined, in the same way as for *CARTEMES*.

MULTIPOL: Magnetic Multipole

The simulation of a multipolar field $\vec{M}$ by *MULTIPOL* proceeds by addition of the dipolar ($\vec{B}1$), quadrupolar ($\vec{B}2$), sextupolar ($\vec{B}3$), octupolar ($\vec{B}4$) decapolar ($\vec{B}5$) and dodecapolar ($\vec{B}6$) components, and of their derivatives up to fourth order. For instance,

$$\vec{M} = \vec{B}1 + \vec{B}2 + \vec{B}3 + \vec{B}4 + \vec{B}5 + \vec{B}6$$
$$\frac{\partial \vec{M}}{\partial X} = \frac{\partial \vec{B}1}{\partial X} + \frac{\partial \vec{B}2}{\partial X} + \frac{\partial \vec{B}3}{\partial X} + \frac{\partial \vec{B}4}{\partial X} + \frac{\partial \vec{B}5}{\partial X} + \frac{\partial \vec{B}6}{\partial X}$$
$$\frac{\partial^2 \vec{M}}{\partial X \partial Z} = \frac{\partial^2 \vec{B}1}{\partial X \partial Z} + \frac{\partial^2 \vec{B}2}{\partial X \partial Z} + \frac{\partial^2 \vec{B}3}{\partial X \partial Z} + \frac{\partial^2 \vec{B}4}{\partial X \partial Z} + \frac{\partial^2 \vec{B}5}{\partial X \partial Z} + \frac{\partial^2 \vec{B}6}{\partial X \partial Z}$$
etc.

The independent components $\vec{B}1$, $\vec{B}2$, $\vec{B}3$, $\vec{B}4$, $\vec{B}5$, $\vec{B}6$ and their derivatives up to the fourth order are calculated as described under *QUADRUPO*, *SEXTUPOL*, *OCTUPOLE*, *DECAPOLE* and *DODECAPO* keywords (see section 2.3.5).

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 (sextupolar, octupolar, decapolar, dodecapolar) component is given by a coefficient $E2$ ($E3$, $E4$, $E5$, $E6$) at entrance, and $S2$ ($S3$, $S4$, $S5$, $S6$) at exit, such that the extent is $\lambda_E * E2$ ($\lambda_E * E3$, $\lambda_E * E4$, $\lambda_E * E5$, $\lambda_E * E6$) at entrance and $\lambda_S * S2$ ($\lambda_S * S3$, $\lambda_S * S4$, $\lambda_S * S5$, $\lambda_S * S6$) 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 $E_i = 0$ ($S_i = 0$), the entrance (exit) fringe field for multipole component i is considered as a sharp edge field. In sharp edge filed 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 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).

Overlapping of fringe fields inside the magnet is treated separately for each component, in the way described in *QUADRUPO*.

Any multipole component $\vec{B}_i$ can be rotated independently by an angle R_{X_i} around the longitudinal X-axis, for the simulation of positioning defects, as well as skewed lenses.

Magnet (mis-)alignement is assured by *KPOS*, with special features allowing some degrees of automatism useful for periodic structures (section 5.6.5).

OCTUPOLE: Octupole magnet (Fig. 24)

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

$$V(X,Y,Z) = \left(G - \frac{G''}{20}(Y^2+Z^2) + \frac{G''''}{960}(Y^2+Z^2)^2\right)(Y^3Z - YZ^3)$$

$$\text{with } G_0 = \frac{B_0}{R_0^3}$$

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

$$B_X = 0$$
$$B_Y = G_0(3Y^2Z - Z^3)$$
$$B_Z = G_0(Y^3 - 3YZ^2)$$

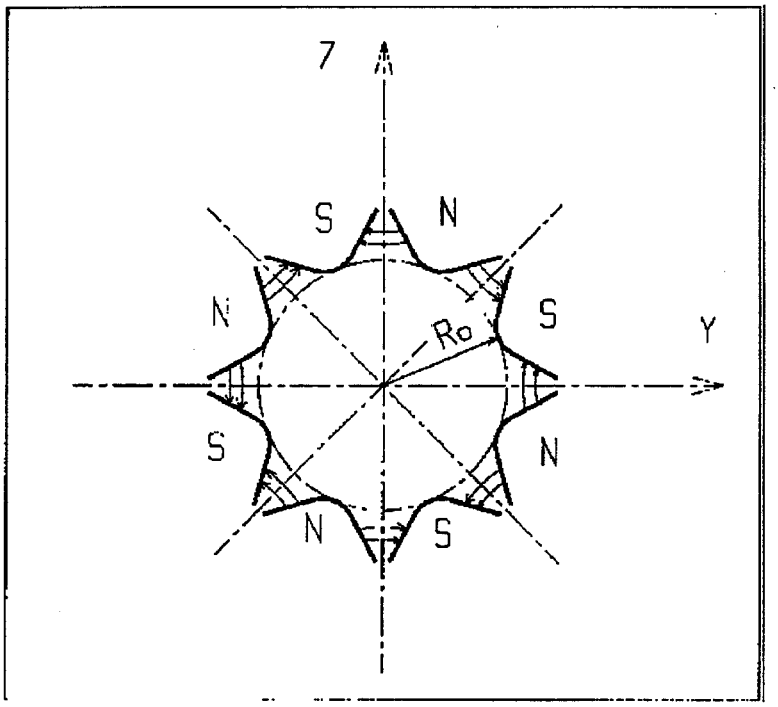


Figure 24: Octupole magnet

POISSON: Read field data from POISSON output

This keyword allows reading a field profile $B(X)$ from a *POISSON* output. Let *FNAME* be the name of this output file (normally, *FNAME* = outpoi.lis); the data are read following the *FORTRAN* statements hereunder

```
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. K's and R's 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 up, 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 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 map in *DIPOLE*).

PS170: Simulation of a Round Shape Dipole Magnet

PS170 is dedicated to a 'rough' simulation of CERN's *PS170* dipole.

The field B_0 is constant inside the magnet, and zero outside. The pole is a circle of radius R_0, centered on X axis. The output coordinates are generated at the distance XL from the entrance (Fig. 25).

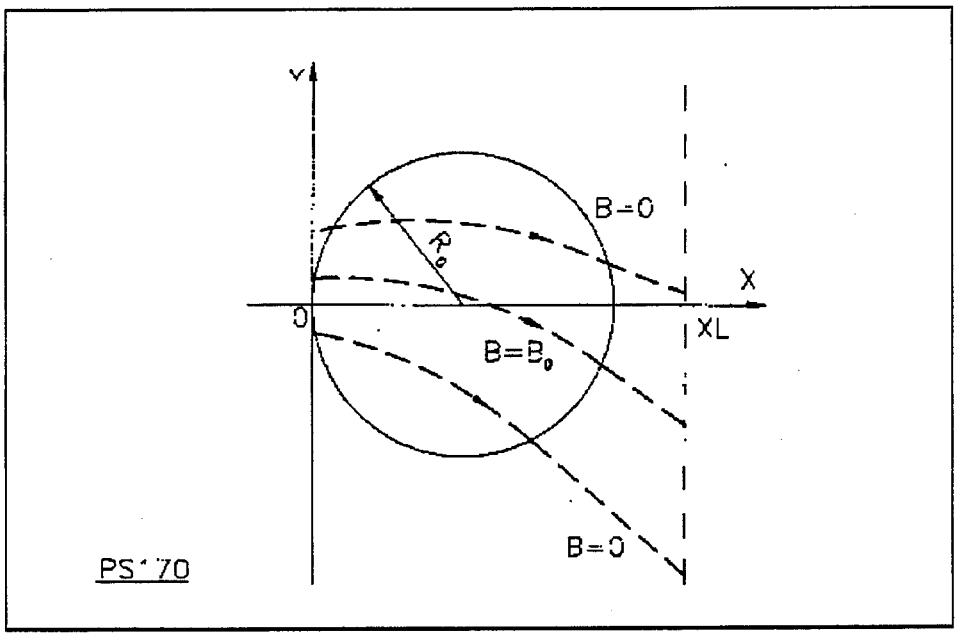


Figure 25: Scheme of the PS170 magnet simulation.

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 from 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 2.3.2).

With option *SEXQUAD*, $U = 0$, while with *QUADISEX*, $U = 1$.

QUADRUPO: Quadrupole Magnet (Fig. 26)

The length of the magnet XL is the distance between the effective field boundaries (EFB). 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 EFB's, in the field fall off regions, will be $\pm X_E$ at the entrance, and $\pm X_S$ at the exit (Fig. 27), by prior and further automatic changes of frame.
In the fringe field regions $[-X_E, X_E]$ and $[-X_S, X_S]$ on both sides of the EFB's, $\vec{B}(X,Y,Z)$ 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 2.3.5) approximated to the 8th order in Y and Z.

$$V(X,Y,Z) = \left(G - \frac{G''}{12}(Y^2+Z^2) + \frac{G''''}{384}(Y^2+Z^2)^2 - \frac{G''''''}{23040}(Y^2+Z^2)^3\right) YZ$$
$$(G' = dG/dX, \quad G'' = d^2G/dX^2, ...)$$

where G is the gradient on axis [16]:

$$G(s) = \frac{G_0}{1+\exp P(s)} \qquad \text{with} \quad G_0 = \frac{B_0}{R_0}$$

and,

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

where, s 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

$$B_X = 0$$
$$B_Y = G_0 Z$$
$$B_Z = G_0 Y$$

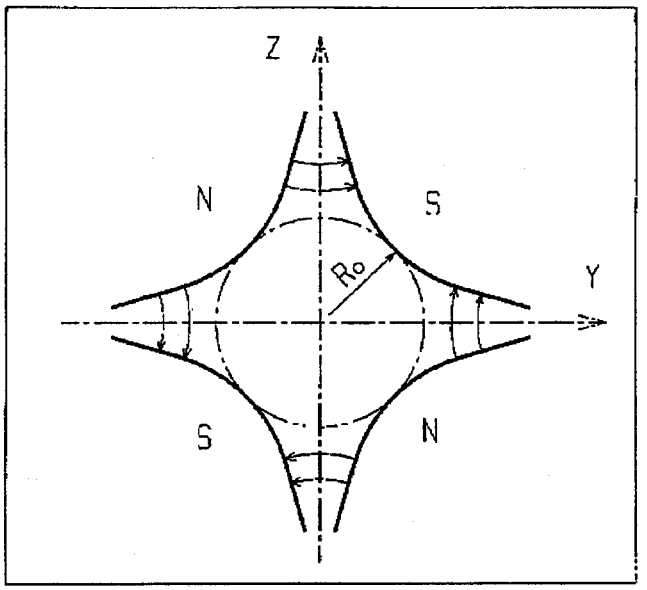


Figure 26: Quadrupole magnet

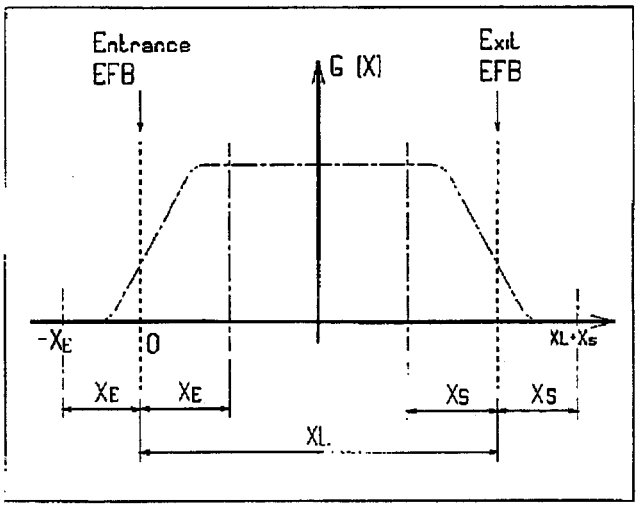


Figure 27: Scheme of the longitudinal field gradient $G(X)$.
(OX) is the longitudinal axis of the reference frame $(0, X, Y, Z)$ of **Zgoubi**. The length of the element is XL, but trajectories are ray-traced from $-X_E$ to $XL + X_S$, by means of prior and further automatic changes of frame.

SEPARA: Wien filter - analytic simulation

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 this simulation 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. 28).

$$X = -R\cos\left(\frac{\omega S}{\beta c} + \epsilon\right) - \frac{\alpha S}{\omega \beta c} + \frac{C_1}{\omega}$$
$$Y = R\sin\left(\frac{\omega S}{\beta c} + \epsilon\right) - \frac{\alpha}{\omega^2} - \frac{C_2}{\omega} + Y_0$$
$$Z = S\sin(P_0) + Z_0$$

where, S is the path length in the separator, $\alpha = -\frac{Ec^2}{\gamma}$, $\omega = -\frac{Bc^2}{m\gamma}$, $C_1 = \beta\sin(T_0)\cos(P_0)$ 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 $\tan\epsilon = (C_2 + \frac{\alpha}{\omega})/C_1$. 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 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

$$B(T) = -\frac{E\left(\frac{V}{m}\right)}{\beta_W \cdot c\left(\frac{m}{s}\right)}$$

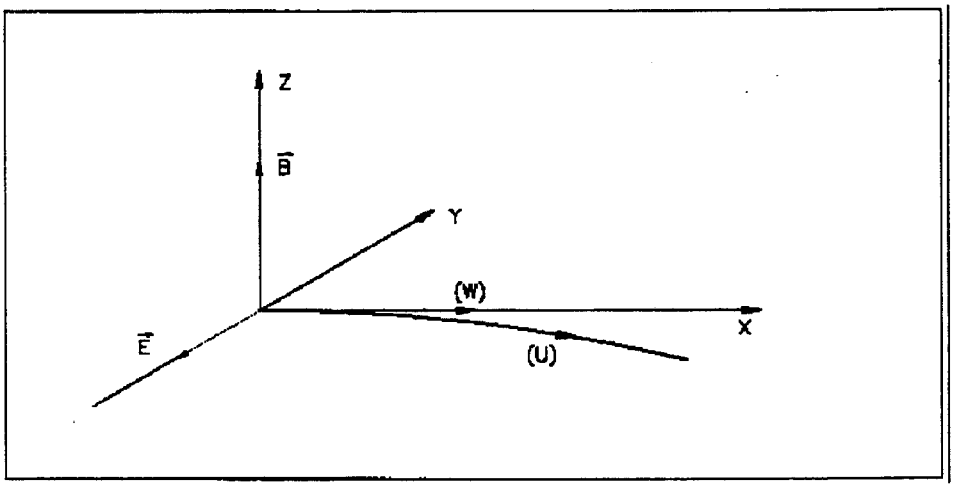


Figure 28: Horizontal separation between a wanted particle, (W), and an unwanted particle, (U). (W) undergoes a linear motion while (U) undergoes a cycloidal motion.

SEXTUPOL: Sextupole Magnet (Fig. 29)

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

$$V(X,Y,Z) = \left(G - \frac{G''}{16}(Y^2+Z^2) + \frac{G''''}{640}(Y^2+Z^2)^2\right)\left(Y^2Z - \frac{Z^3}{3}\right)$$

$$\text{with } G_0 = \frac{B_0}{R_0^2}$$

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

$$B_X = 0$$
$$B_Y = 2G_0YZ$$
$$B_Z = G_0(Y^2 - Z^2)$$

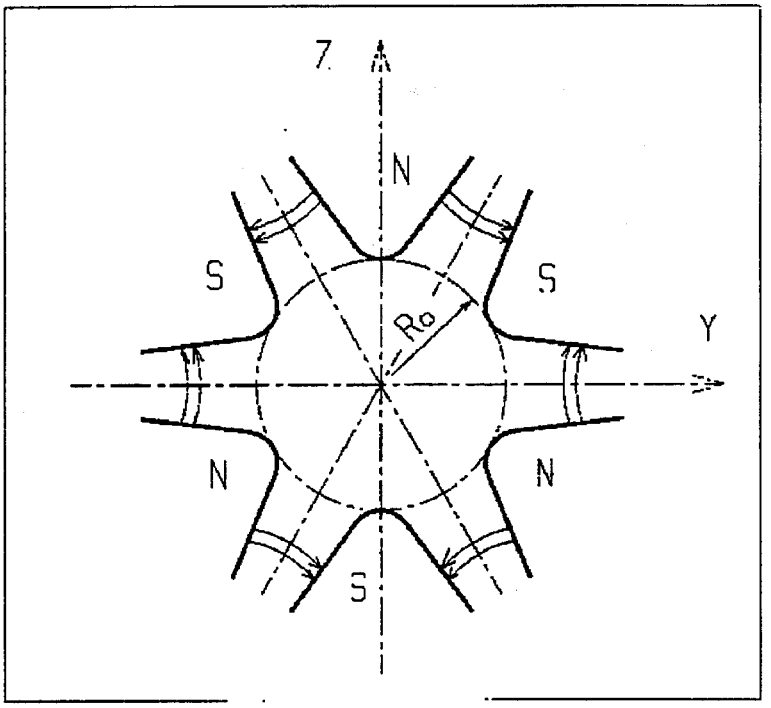


Figure 29: Sextupole magnet

SOLENOID: Solenoid (Fig. 30)

The solenoidal magnet has an effective length XL, a mean radius R_0 and an asymptotic field $B_0 = \mu_0 NI$ (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. 30).

The field $\vec{B}(X,r)$, $r = (Y^2 + Z^2)^{1/2}$, and its derivatives up to the second order with respect to X, Y or Z are obtained after the method proposed in ref. [22], that involves the three complete elliptic integrals K, E and Π. These are calculated with the algorithm proposed in the same reference. Their derivatives are calculated by means of recursive relations [23].

This analytical model for the solenoidal field allows simulating an extended range of coil geometry provided that the coil thickness is small enough compared to the mean radius R_0.

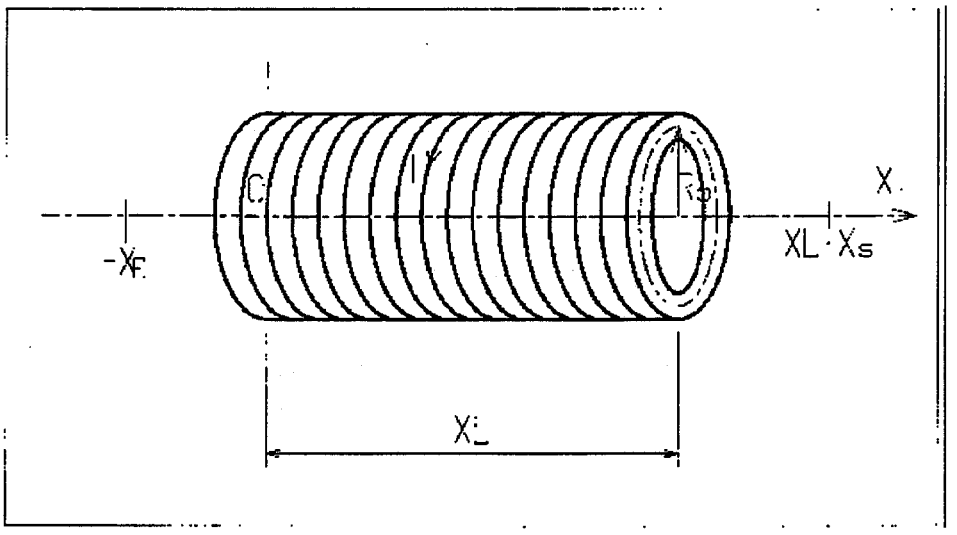


Figure 30: Solenoidal magnet.

TOSCA: 2-D or 3-D Cartesian Uniform Mesh Field Map

TOSCA is dedicated to the reading and treatment of 2-D or 3-D Cartesian mesh field maps as delivered by the *TOSCA* magnet computer code standard output.

The total number of field data files to be read is given by the parameter IZ that appears in the data list following the keyword. Each file contains the field components B_X, B_Y, B_Z on an (X, Y) mesh at a given Z coordinate. $IZ = 1$ for 2-D maps, and in this case B_X and B_Y are assumed to be zero. For 3-D maps with mid-plane symmetry, IZ should be greater than 1, and thus, the first data file, whose name follows in the data list, contains the median plane field (assuming $Z = 0$ and $B_X = B_Y = 0$), while the next files contain the next maps in increasing Z order. For 3-D maps without mid-plane symmetry assumption, IZ should be odd and negative, and thus, the total number of maps (whose names follow in the data list) is $|IZ|$, while the map number $[IZ/2] + 1$ is the $Z = 0$ one.

The field map data files should be formatted following the *FORTRAN* reading sequence below.

```
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(I,J,K), BZ(I,J,K), BX(I,J,K)
              ELSE
        READ(NL,100) Y(J), Z(K), X(I), BY(I,J,K), BZ(I,J,K), BX(I,J,K)
100             FORMAT(1X,6E11.2)
              ENDIF
1       CONTINUE
```

where, IX (JY, KZ) is the number of longitudinal (transverse, vertical) nodes of the 3-D uniform mesh. For binary files, FNAME must begin with 'B_ ', 'BINARY' will then be '.TRUE.'.

The field $\vec{B} = (B_X, B_Y, B_Z)$ is normalized by means of *BNORM*in a similar way as in *CARTEMES*.

At each step of the trajectory of a particle inside the map, the field and its derivatives are calculated by means of a second order polynomial fit with a $3 \times 3 \times 3$-point parallelipipedic grid, as described in section 2.4.4.

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*.

TRANSMAT: Matrix Transfer

TRANSMAT performs a matrix transfer 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 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 [14]. 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

This procedure transports particles into a new frame by translation and rotation. Effect on spin tracking, particle decay and gas-scattering are taken into account (but not on synchrotron radiation).

UNIPOT: Unipotential 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. 31). The model of electric potential along the axis is [24]

$$V(x) = \frac{V2 - V1}{2\omega D}\left[\ell n\,\mathrm{ch}\frac{\omega\left(x + \frac{X2}{2} + D\right)}{R_0}\,\mathrm{ch}\frac{\omega\left(x + \frac{X2}{2}\right)}{R_0} + \ell n\,\mathrm{ch}\frac{\omega\left(x - \frac{X2}{2} - D\right)}{R_0}\,\mathrm{ch}\frac{\omega\left(x - \frac{X2}{2}\right)}{R_0}\right]$$

(x = distance from the center of the central electrode; $\omega = 1{,}318$; ch = hyperbolic cosine), from which the field $\vec{E}(X, Y, Z)$ and its derivatives are deduced following the procedure described in section 2.5.2.
Use *PARTICUL* for the definition of the particle mass and charge

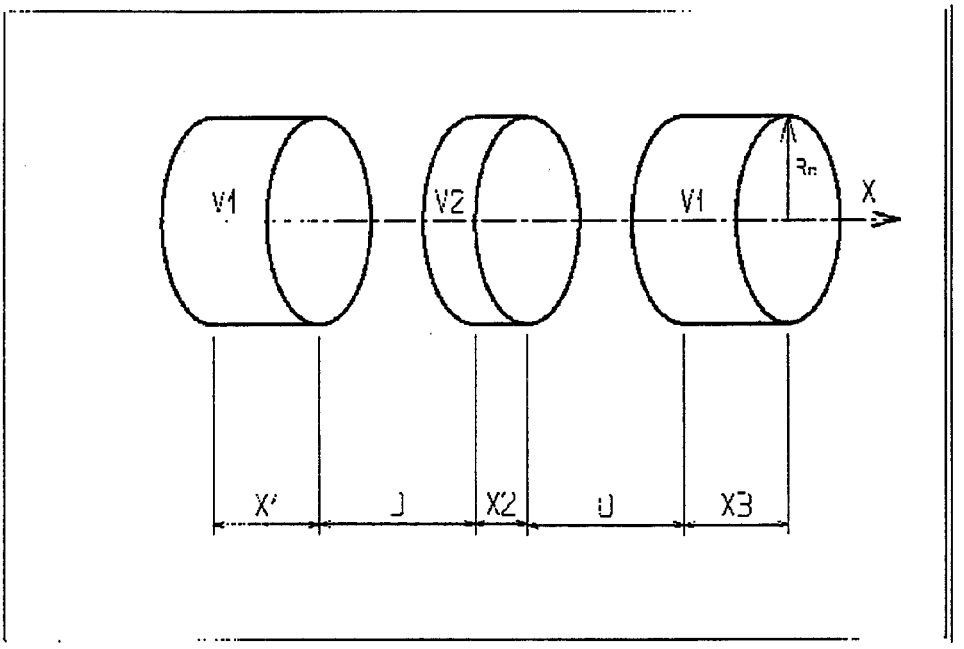


Figure 31: Three-electrode cylindrical unipotential lens.

VENUS: Simulation of a Rectangular 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. 32).

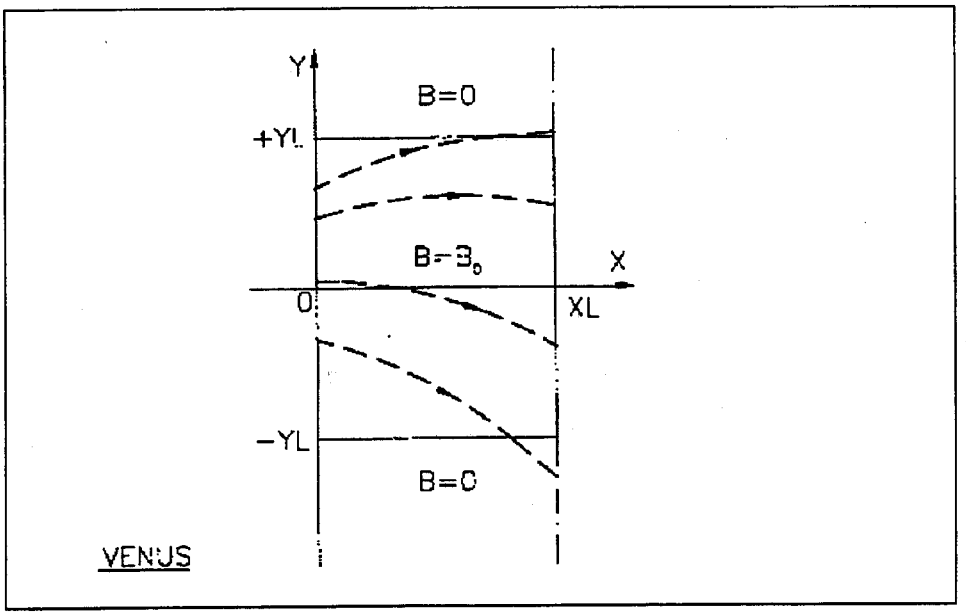


Figure 32: Scheme of *VENUS* rectangular dipole.

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. 28). It must be preceded by *PARTICUL* for the definition of the particle mass and charge.

The length XL of the element is the distance between its entrance and exit EFB's. The electric and magnetic field intensities E_0 and B_0 in the central, uniform field region, normally verify 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 changes 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 [16]

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

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 focusing effects of the non zero longitudinal electric field component.

YMY: Reverse Signs of Y and Z Axis

YMY performs a 180° rotation of particle coordinates with respect to the X-axis, as shown in Fig. 33. This is done by means of a change of sign of Y and Z axes, and therefore coordinates, as follows

$$Y2 = -Y1, \quad T2 = -T1, \quad Z2 = -Z1 \quad \text{and} \quad P2 = -P1$$

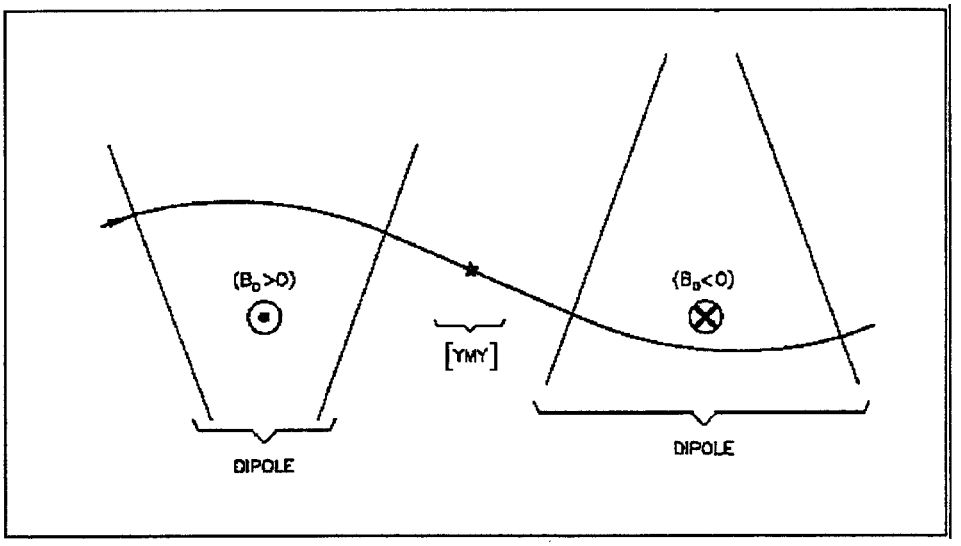


Figure 33: The use of YMY in a sequence of two identical dipoles of opposite signs.

5.5 Output Procedures

These procedures are dedicated to the printing of particle coordinates, histograms, spin coordinates, etc. They may be called for at any spot in the data pile.

CLORB: Beam Centroid Path; Closed Orbit

CLORB computes the beam centroid path, from average value of particle coordinates as observed at *LABEL*'ed keywords.

In conjunction with *REBELOTE*, this procedure computes the closed orbit in the periodic structure delimited with *REBELOTE*, by the same means.

The *LABEL* list of concern constitutes the information contained in the data record that follows the keyword *CLORB*.

FAISCEAU, FAISCNL, FAISCNLA: Print/Store Particle Coordinates

FAISCEAU can be introduced anywhere in a structure. It produces a print of initial and actual coordinates of the particles at the location where it stands, together with their tagging indices and letters, following the same format as for *FAISCNL* (except for *SORT(I)* which is not printed) .

FAISCNL has a similar effect, except that the information is stored in a dedicated file *FNAME* (standard name is *FNAME* = 'zgoubi.fai' for post-processing with **zgplot**). 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 below

```
  OPEN (UNIT = NL, FILE = FNAME, STATUS = 'NEW')
  DO  1  I=1, IMAX
    WRITE(NL,100) LET(I),IEX(I),(FO(J,I),J=1,6),(F(J,I),J=1,6),I(I),IREP(I),SORT(I),DUM,DUM,RET(I), DPR(I), IPASS
100     FORMAT(1X, A1, 1X, I2, 1P, 6E16.8, /, 6E16.8, 2I3, /,1X, 5E16.8, I6)
    1     CONTINUE
```

The meaning of these parameters 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 eventually been stopped (see *CHAMBR* or *COLLIMA*)
DUM	:	dummy
$RET(I)$, $DPR(I)$	:	synchrotron phase space coordinates; RET =phase (radian), DPR = momentum dispersion (MeV/c) (see *CAVITE*).
$IPASS$	:	turn number (see *REBELOTE*)

FAISCNLA 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 proper to the elements of the data file at the exit of which the print should occur; if there is no label, the print occurs by default at the location of *FAISCNLA*; if there are labels the print occurs right downstream the optical element wearing those labels (and no longer at the *FAISCNLA* location). The next data line gives a parameter *IP*: printing will occur every *IP* other pass, if using *REBELOTE* with *NPASS* $\geq IP - 1$. For instance the data list

FAISCNLA
zgoubi.fai *HPCKUP* *VPCKUP*
12

will result in output prints into zgoubi.fai, every 12 other pass, each time elements of the zgoubi.dat data list labeled either *HPCKUP* or *VPCKUP* are encountered.

FOCALE, IMAGE[S]: Coordinates and Beam Dimension, 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:

$$\begin{aligned} &FOCALE: \quad X = XL \\ &IMAGE[S]: \quad X = -\frac{\sum_{i=1}^{N} Y_i * tgT_i - \left(\sum_{i=1}^{N} Y_i * \sum_{i=1}^{N} tgT_i\right)/N}{\sum_{i=1}^{N} tg^2T_i - \left(\sum_{i=1}^{N} tgT_i\right)^2/N} \\ &\qquad\qquad Y = Y_1 + X * \mathrm{tg}T_1 \end{aligned}$$

  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

$$YM = \frac{1}{N}\sum_{i=1}^{N}(Y_i + X\mathrm{tg}T_i) - Y = \frac{1}{N}\sum_{i=1}^{N} Y\, M_i$$

- FWHM of the image (*IMAGE[S]*) or of the beam (*FOCALE*), and total width, respectively, W and WT

$$\begin{aligned} W &= 2.35\left(\frac{1}{N}\sum_{i=1}^{N} Y\, M_i^2 - Y\, M^2\right)^{\frac{1}{2}} \\ WT &= \max(YM_i) - \min(YM_i) \end{aligned}$$

FOCALEZ, IMAGE[S]Z: Coordinates and Beam Dimensions, 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.

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 actual 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 variable. If it is set equal to 'S', only secondary particles will be histogrammed. If it is set equal to 'P', then only primary particles will be histogrammed. For no discrimination between S-econdary and P-rimary 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 spots in a structure. This is also useful when piling up problems in an 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.

IMAGE[S][Z]: Localization and size of Vertical Waist

See FOCALE[Z].

MATRIX: Calculation of Transfer Coefficients

MATRIX causes the calculation of the transfer coefficients of the structure, at the spot where it is introduced in the structure, or at the closest horizontal focus. 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 option *IFOC*, *MATRIX* also delivers the Twiss matrix and tune numbers in the hypothesis of a periodic structure.

Depending on the value of option *IORD*, different procedures follow

- If $IORD = 0$, *MATRIX* is inhibited (equivalent to *FAISCEAU*, whatever *IFOC*).
- If $IORD = 1$, the first order transfer matrix $[R_{ij}]$ is calculated, from a third order expansion of the coordinates, for instance

$$Y^+ = \left(\frac{Y}{T_0}\right) T_0 + \left(\frac{Y}{T_0^2}\right) T_0^2 + \left(\frac{Y}{T_0^3}\right) T_0^3$$
$$Y^- = -\left(\frac{Y}{T_0}\right) T_0 + \left(\frac{Y}{T_0^2}\right) T_0^2 - \left(\frac{Y}{T_0^3}\right) T_0^3$$

which gives, neglecting third order terms

$$R_{11} = \left(\frac{Y}{T_0}\right) = \frac{Y^+ - Y^-}{2T_0}$$

- If $IORD = 2$, fifth order Taylor expansions are used for the calculation of the first order transfer matrix $[R_{ij}]$ and the second order matrix $[T_{ijk}]$. Other higher order coefficients are also calculated.

The next option, *IFOC*, acts as follows

- If $IFOC = 0$, the transfer coefficients are calculated at the position of *MATRIX*, and with respect to particle 1 taken as a reference (for instance, Y^+ and T^+ above are defined for particle I as $Y^+ = Y^+(I) - Y(1)$, and $T^+ = T^+(I) - T(1)$).
- If $IFOC = 1$, the transfer coefficients are calculated at the horizontal focus which is the closest to *MATRIX* (determined automatically), while the reference direction is that of particle 1 (for instance, Y^+ is defined for particle I as $Y^+ = Y^+(I) - Y_{\text{focus}}$, and T^+ is defined as $T^+ = T^+(I) - T(1)$).
- 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.
- If $IFOC = 10 + NPeriod$, *MATRIX* calculates the transfer coefficients of the structure, assuming that it is *NPeriod*-periodic, and deduces the corresponding Twiss matrix and tune numbers. No change of reference is performed for this calculation.

The object necessary for the calculation of $[R_{ij}]$ with $IORD = 1$ may be generated automatically by means of *OBJET* with option $KOBJ = 5$. When using $IORD = 2$, the object may be generated automatically with *OBJET* and $KOBJ = 6$.

PLOTDATA: Coordinate Output for PLOTDATA Graphic Software [25]

To be documented.

SPNPRNL, SPNPRNLA, SPNPRT: Print/Store Particle Spin Coordinates

SPNPRNL has the same effect as *SPNPRT* (see below), except that the information is stored in a dedicated file *FNAME* (standard is *FNAME* = 'zgoubi.spn' for post-processing with **zgplot**). The data are formatted and ordered according to the *FORTRAN* sequence below

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'NEW')
DO  1  I=1, IMAX
    WRITE (NL,100) LET(I), IEX(I), (SI(J,I)J=1,4), (SF(J,I),J=1,4), GAMMA, I
 100   FORMAT(1X, A1, I2, 1P, 8E15.7, /, E15.7, 2I3, I6)
  1     CONTINUE
```

The meaning of these parameters is the following

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*)

SPNPRNLA has an effect similar to *SPNPRNL*, with one more feature. The line next to *FNAME* gives a parameter IP printing will occur every IP other pass, when using *REBELOTE* with $NPASS \geq IP - 1$.

SPNPRT can be introduced anywhere in a structure. It produces a listing (into zgoubi.res) of the initial and actual coordinates and modulus of the spin of the *IMAX* particles, at the location where it stands, together with their Lorentz factor γ, following the format detailed above. The mean values of the spin components are also printed.

5.6 Complementary Features

5.6.1 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 34.

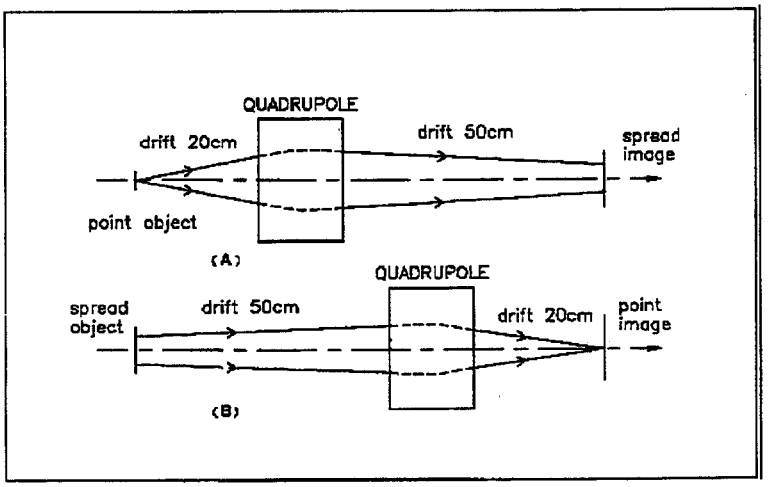


Figure 34: A. Regular forward ray-tracing, from object to image.
B. Same structure, with backward ray-tracing from image to object: negative integration step XPAS is used in the quadrupole.

5.6.2 Checking Fields and Trajectories in Magnets

In all magnetic elements, an option index IL is available. It is normally set to 0 and in this case has no effect.
$IL = 1$ causes a print in zgoubi.res of particle coordinates and field along trajectories in the magnet. In the meantime, a calculation and summation of the values of $\vec{\nabla} \cdot \vec{B}$, $\vec{\nabla} \times \vec{B}$ and $\nabla^2 \vec{B}$ at all integration steps is performed, which allows a check of the behavior of $\vec{B}$ in field maps (all these derivatives should normally be zero).
$IL = 2$ causes a print of particle coordinates and other informations in zgoubi.plt which can further be processed with **zgplot**[2].

When dealing with maps (e.g., *CARTEMES*, *ELREVOL*), another option index IC is available. It is normally set to 0 and in this case has no effect.
$IC = 1$ causes a print of the field map in zgoubi.res.
$IC = 2$ will cause a print of field maps in zgoubi.map which can further be processed with **zgplot**[2].

5.6.3 Labeling keywords

Keywords in **Zgoubi** data file zgoubi.dat can be *LABEL*'ed, for the purpose of the execution of such procedures as *CLORB*, *FAISCNL[A]* , *SCALING*.
Each keyword accepts two *LABEL*'s, of which the first one is used for the above mentioned purpose.

[2]See Part D of the Guide.

5.6.4 Multiturn tracking in circular machines

Multiturn tracking in circular machines can be performed by means of the keyword *REBELOTE*, put at the end of the optical structure with its argument *NPASS*+1 being the number of turns to be performed. In order that the *IMAX* particles of the beam start a new turn with the coordinates they have reached at the end of the previous one, the option $K = 99$ has to be specified in *REBELOTE*.

Synchrotron acceleration can be simulated, following the procedure below

- *CAVITE* appears at the end of the structure (before *REBELOTE*), with option *IOPT*= 1
- the R.F. frequency of the cavity is given a timing law by means of *SCALING*, family *CAVITE*
- the magnets are given the same timing law $B\rho(T)$, (where $T = 1$ to *NPASS*+1 is the turn number) by means of *SCALING*.

Eventually some families of magnets may be given a law which does not follow $B\rho(T)$, for the simulation of special processes (e.g. fast crossing of spin resonances with independent families of quadrupoles).

5.6.5 Positioning of Magnets and field maps

The last record in most magnets and field maps is the positioning flag *KPOS*, followed by the parameterss *XCE*, *YCE* for translation and *ALE* for rotation. The positioning works in two different ways, depending on whether they are defined in Cartesian (X, Y, Z) coordinates (e.g., *QUADRUPO, TOSCA*), or polar (R, θ, Z) coordinates (*DIPOLE*).

Cartesian Coordinates:

If $KPOS = 1$, the X-axis of the element coincides with the X-axis of the incoming reference.
If $KPOS = 2$, the shifts *XCE* and *YCE*, and the tilt angle *ALE* are taken into account, for the positioning of the element with respect to the incoming reference, as shown in Fig. 35. $KPOS = 2$ can also be used to simulate a misalignment. The effect is equivalent to a *CHANGREF* transformation placed right upstream the magnet, followed by the reverse transformation right downstream.
$KPOS = 3$ option is available for some magnets (e.g., *BEND*, *MULTIPOL*); it positions automatically the magnet in the following way, convenient for periodic structures. It is effective only if a non zero dipole component B_1 is present; entrance and exit frames are shifted by *YCE* (*XCE* is not used) and tilted w.r.t. the magnet by an angle

- either ALE/2 if ALE$\neq$0
- or by half the deviation $\theta/2$ such that $L = 2\frac{BORO}{B1}\sin(\theta 2)$ if ALE=0 (L = magnet length, $BORO$ = reference rigidity as defined in *OBJET*). This is equivalent to the sequence *CHANGREF(0,0,-θ/2)*, *CHANGREF(0,YCE,0)* right upstream the magnet, followed by *CHANGREF(0,-YCE,-θ/2)*.

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 momentum 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. 36; see *DIPOLE* and *POLARMES*).
If $KPOS = 2$, the map is positioned in such a way that the incoming particle will enter it at radius RE with angle TE. The reference frame of **Zgoubi** is positioned in a similar way with respect to the map, at the exit face, by means of the two parameters RS (radius) and TS (angle) (see Fig. 10A.).

5.6.6 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* = b.fffE10 in order to allow two different step sizes in the uniform part of the filed (the magnet body) and in the field fall-off regions. b is an arbitrary integer and fff is a 3-digit integer; they give the number of steps respectively in the body and fringe field regions. For instance 120.012E10 requests 120 steps in the body and 12 in the fringe field regions. The maximum allowed value for fff is 999 steps.

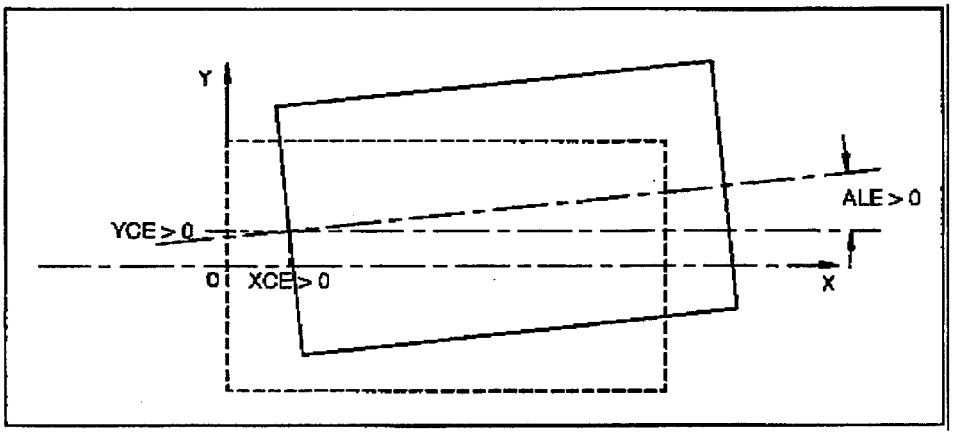


Figure 35: Positioning of a Cartesian coordinate optical element when *KPOS*= 2.

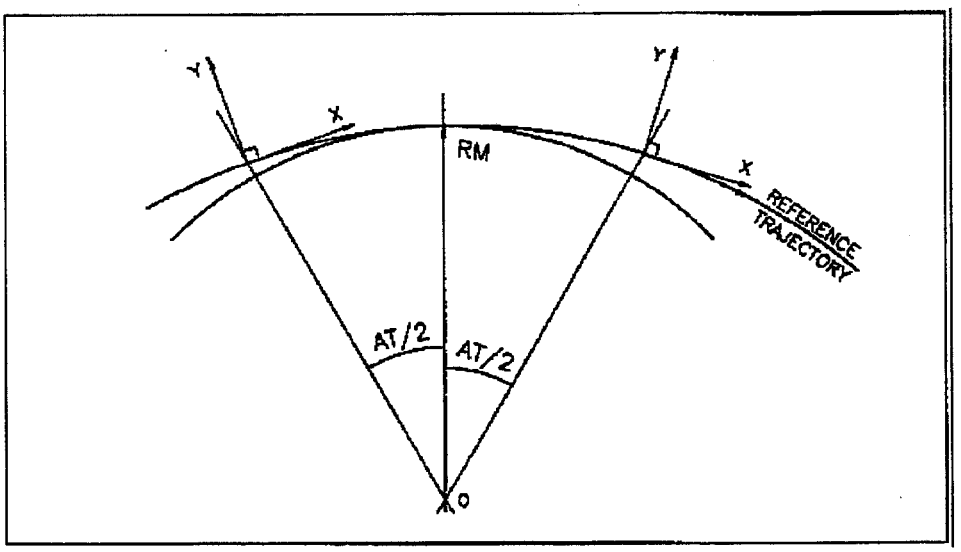


Figure 36: Positioning of a polar field map when *KPOS*= 1.

5.6.7 Ray-tracing of an arbitrarily large number of particles

Monte Carlo multiparticle simulations involving an arbitrary number of particles can be performed by means of *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.

5.6.8 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 take negative values, as follows

- 1 : the trajectory happened to wander outside the limits of a field map
- 2 : too many integration steps in a field map
- 3 : deviation happened to exceed $\frac{\pi}{2}$ in an optical element
- 4 : stopped in chamber walls (by the procedures *CHAMBR*, *COLLIMA*)
- 5 : too many iterations in subroutine *DEPLA*

Only in the case $IEX = -1$ will the particle not be stopped.

5.6.9 Negative rigidity

Zgoubi can handle negative rigidities $B\rho = p/q$. This is equivalent to considering either particles of negative charges ($q < 0$), or counter going particles ($p < 0$), or virtually reversed fields (w.r.t. the field sign that shows in the 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* and *MCOBJET*.

FERMILAB-TM-2010

PART B

Keywords and input data formatting

Glossary of keywords

AIMANT	Generation of a dipole magnet 2-D map	121
AUTOREF	Automatic transformation to a new reference frame	125
BEND	Bending magnet	126
BINARY	BINARY/FORMATTED data converter	127
BREVOL	1-D uniform mesh magnetic field map	128
CARTEMES	2-D cartesian uniform mesh magnetic field map	129
CAVITE	Accelerating cavity	131
CHAMBR	Long transverse aperture limitation	132
CHANGREF	Transformation to a new reference frame	133
CIBLE	Generate a secondary beam from target interaction	134
CLORB	Beam Centroid Path; Closed Orbit	135
COLLIMA	Collimator	136
DECAPOLE	Decapole magnet	137
DIPOLE	Generation of a dipole magnet 2-D map	138
DODECAPO	Dodecapole magnet	140
DRIFT	Field free drift space	141
EBMULT	Electro-magnetic multipole	142
EL2TUB	Two-tubes electrostatic lense	144
ELMULT	Electric multipole	145
ELREVOL	1-D uniform mesh electric field map	146
END	End of input data list ; see FIN	148
ESL	Field free drift space	141
FAISCEAU	Print particle coordinates	147
FAISCNL	Store particle coordinates into file *FNAME*	147
FAISCNLA	Store coordinates every IP other pass at labelled elements	147
FIN	End of input data list	148
FIT	Fitting procedure	149
FOCALE	Particle coordinates and horizontal beam dimension at distance XL	150
FOCALEZ	Particle coordinates and vertical beam dimension at distance XL	150
GASCAT	Gas scattering	151
HISTO	1-D histogram	152
IMAGE	Localization and size horizontal waist	153
IMAGES	Localization and size of horizontal waists	153
IMAGESZ	Localization and size of vertical waists	153
IMAGEZ	Localization and size of vertical waist	153
MAP2D	2-D cartesian uniform mesh magnetic field map without symmetry	154
MATRIX	Calculation of transfer coefficients	155
MCDESINT	Monte-Carlo simulation of in-flight decay	156
MCOBJET	Monte-Carlo generation of a 3-D object	157
MULTIPOL	Magnetic multipole	160
OBJET	Generation of an object	161
OBJETA	Object from Monte-Carlo simulation of decay reaction	163
OCTUPOLE	Octupole magnet	164
ORDRE	Higher order Taylor expansions in lenses	165
PARTICUL	Particle characteristics	166
PLOTDATA	Intermediate outputs for the *PLOTDATA* computer graphic software	167
POISSON	Read field data from *POISSON* output	168
POLARMES	2-D polar mesh field map	169
PS170	Simulation of a round shape dipole magnet	170
QUADISEX	Sharp edge magnetic multipoles	171
QUADRUPO	Quadrupole magnet	172

REBELOTE Jump to the beginning of Zgoubi input data file 174
RESET Reset counters and flags 175
SCALING Time scaling of power supplies and R.F. 176
SEPARA Wien Filter - analytic simulation 177
SEXQUAD Sharp edge magnetic multipole 178
SEXTUPOL Sextupole magnet 179
SOLENOID Solenoid 180
SPNPRNL Store spin coordinates into file *FNAME* 181
SPNPRNLA Store spin coordinates every *IP* other pass 181
SPNPRT Print spin coordinates 181
SPNTRK Spin tracking 182
SYNRAD Synchrotron radiation 183
TARGET Generate a secondary beam from target interaction ; see CIBLE 134
TOSCA 2-D and 3-D cartesian uniform mesh field map 184
TRANSMAT Matrix transfer 185
TRAROT Translation-Rotation of the reference frame 186
UNIPOT Unipotential electrostatic lense 187
VENUS Simulation of a rectangular dipole magnet 188
WIENFILT Wien filter 189
YMY Reverse signs of Y and Z axes 190

Optical elements versus keywords

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

Field map procedures are also cataloged; in most cases an adequate field map can be used for simulating these elements.

MAGNETIC ELEMENTS

Decapole	DECAPOLE, MULTIPOL
Dipole	AIMANT, BEND, DIPOLE, MULTIPOL, QUADISEX
Dodecapole	DODECAPO, MULTIPOL
Multipole	MULTIPOL, QUADISEX, SEXQUAD
Octupole	OCTUPOLE, MULTIPOL, QUADISEX, SEXQUAD
Quadrupole	QUADRUPO, MULTIPOL, SEXQUAD
Sextupole	SEXTUPOL, MULTIPOL, QUADISEX, SEXQUAD
Skewed multipoles	MULTIPOL
Solenoid	SOLENOID

Field maps

1-D, cylindrical symmetry	BREVOL
2-D, mid-plane symmetry	CARTEMES, POISSON, TOSCA
2-D, no symmetry	MAP2D
3-D	TOSCA

ELECTRIC ELEMENTS

Decapole	ELMULT
Dipole	ELMULT
Dodecapole	ELMULT
Multipole	ELMULT
Octupole	ELMULT
Quadrupole	ELMULT
R.F. cavity	CAVITE
Sextupole	ELMULT
Skewed multipoles	ELMULT
2-tube (bipotential) lense	EL2TUB
3-tube (unipotential) lense	UNIPOT

Field maps

1D, cylindrical symmetry	ELREVOL

ELECTROMAGNETIC ELEMENTS

Decapole	EBMULT
Dipole	EBMULT
Dodecapole	EBMULT
Multipole	EBMULT
Octupole	EBMULT
Quadrupole	EBMULT
Sextupole	EBMULT
Skewed 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 may therefore need be enclosed between quotes (e.g., '*DIPOLE*').

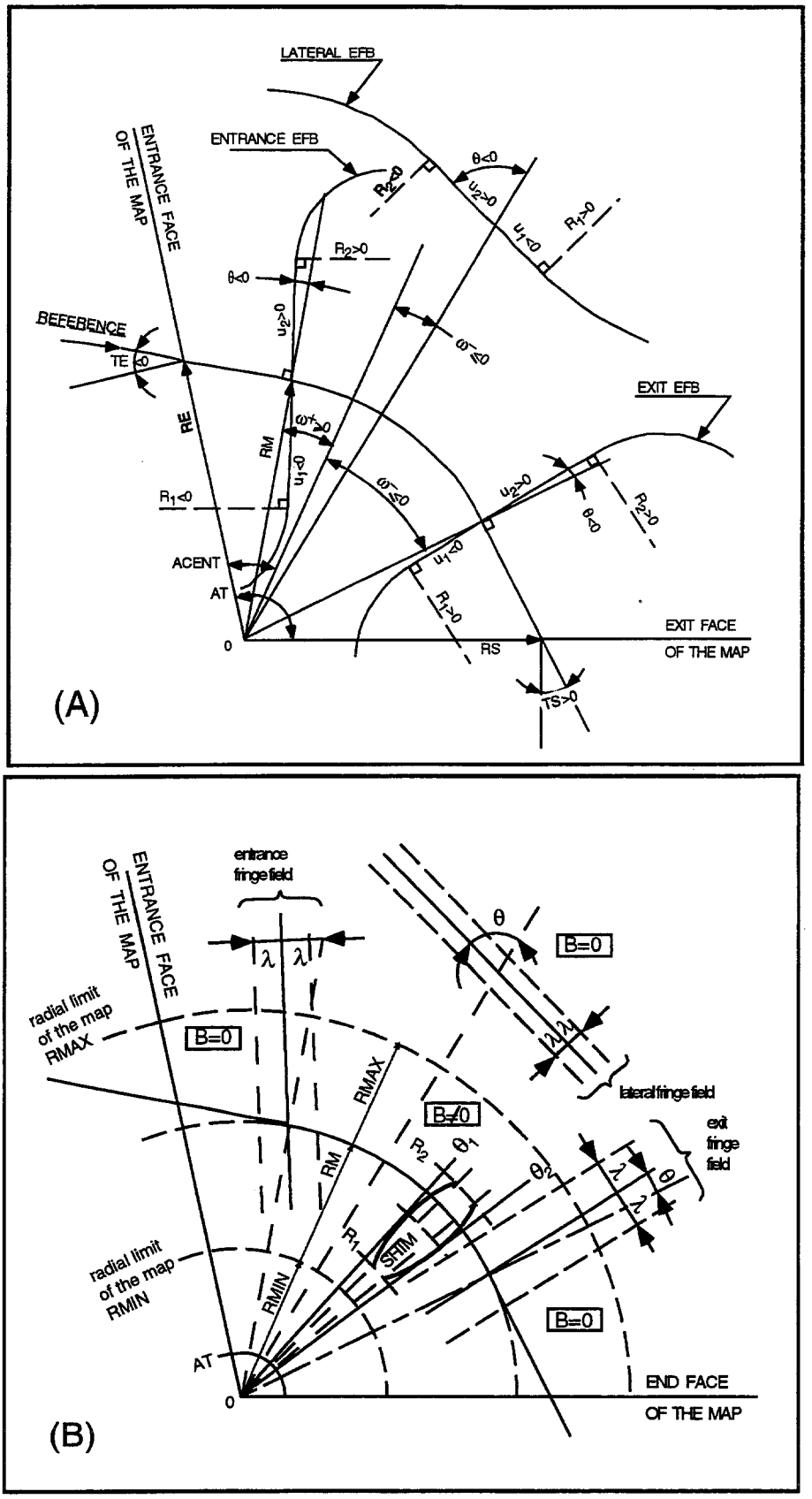
Text string data such as comments or file names, are read by formatted READ statements. Therefore no quotes are needed. Numerical variables and indices are read by unformatted READ. It may therefore be necessary that integer variables be assigned an integer value.

In the following tables

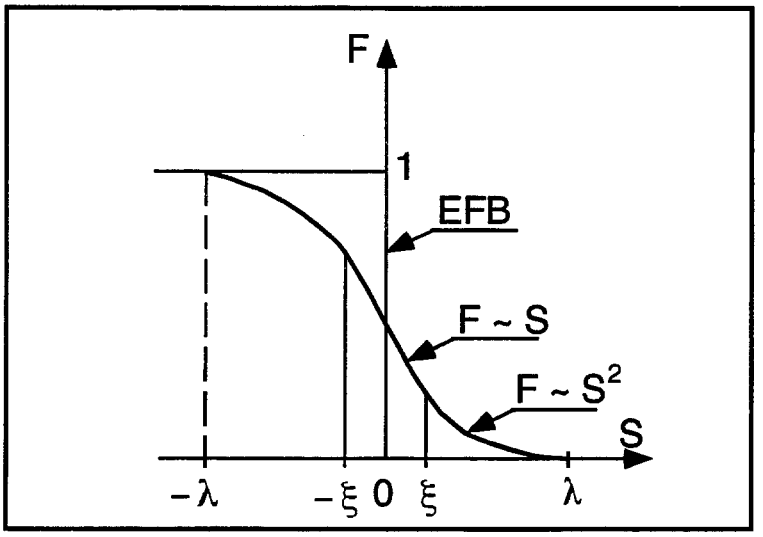
- the first column states the input numerical variables, indices and text strings,
- the second column gives brief explanations,
- the third column gives the units or ranges of the input variables and indices,
- the fourth column indicates whether the inputs are integers (I), reals (E) or text strings (A). For example, 'I, 3*E' means that one integer followed by 3 reals must be entered. 'A80' means that a text string of maximum 80 characters must be entered.

AIMANT	**Generation of a dipole magnet 2-D map** $B_Z = \mathcal{F} B_0 \left(1 - N\left(\frac{R-RM}{RM}\right) + B\left(\frac{R-RM}{RM}\right)^2 + G\left(\frac{R-RM}{RM}\right)^3\right)$		
NFACE, IC, IL	Number of field boundaries $IC = 1, 2$: print field map $IL = 1, 2$: print field and coordinates on trajectories	2-3, 0-2, 0-2	3*I
IAMAX, IRMAX	Azimuthal and radial number of nodes of the mesh	$\leq 400, \leq 200$	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 EFB's; mean radius; minimum and maximum radii	2*deg, 3*cm	5*E
	ENTRANCE FIELD BOUNDARY		
λ, ξ	Fringe field extent; index for fringe field as follows: if $\xi \geq 0$: second order type fringe field with linear variation over 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 + order 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 $\mid U_{1,2} \mid = \infty$ when $R_{1,2} = \infty$) (Note : $\lambda = 0$, ω^+ = ACENT and $\theta = 0$ for <u>sharp edge</u>)	2*deg, 4*cm	6*E
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
λ, ξ	Fringe field parameters	cm, (cm)	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	Positioning and shape of the exit EFB (Note : $\lambda = 0$, ω^- =-AT+ACENT and $\theta = 0$ for <u>sharp edge</u>)	2*deg, 4*cm	6*E

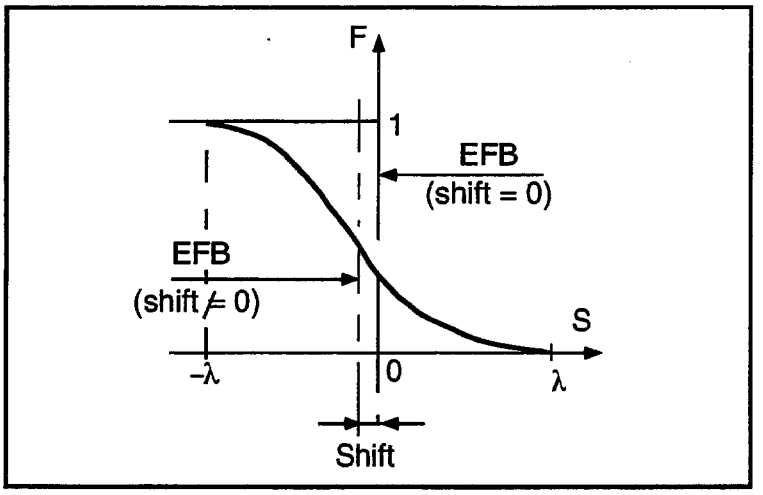
if NFACE = 3	LATERAL FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY) Next 3 records *only* if *NFACE* = 3		
λ, ξ	Fringe field parameters	cm, (cm)	2*E
NC, $C_0 - C_5$, shift		0-6, 6* no dim., cm	I, 7*E
ω^-, θ, R_1, U_1, U_2, R_2, $RM3$	Positioning and shape of the lateral EFB; RM3 is the radial position on azimut *ACENT*	2*deg, 5cm	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 $\geq$ 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	Interpolation polynomial order: 2 = second order, 9-point grid 25 = second order, 25-point grid 4 = fourth order, 25-point grid	2, 4 or 25	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



A: Parameters used to define the field map and geometric boundaries.
B: Parameters used to define the field map and fringe fields.



Second order type fringe field.

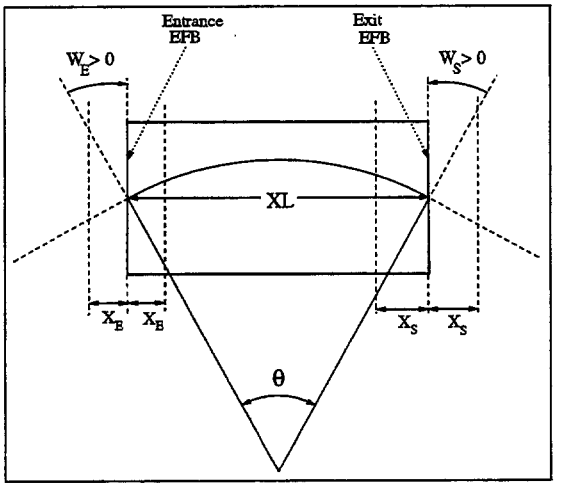


Exponential type fringe field.

AUTOREF **Automatic transformation to a new reference frame**

I	1: Equivalent to *CHANGREF* ($XCE = 0$, $YCE = Y(1)$, $ALE = T(1)$)	1-2	I
	2: Equivalent to *CHANGREF* (XW, YW, $T(1)$), with (XW, YW) being the position of the intersection (waist) of particles 1, 4 and 5 (useful with *MATRIX*, for automatic positionning of the first order focus)		
	3: Equivalent to *CHANGREF* (XW, YW, $T(I1)$), with (XW, YW) being the position 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)		
if $I = 3$	Next record only if $I = 3$		
$I1$, $I2$, $I3$	Three particle numbers	3*(1-200)	3*I

BEND	**Bending magnet**		
IL	$IL = 1, 2$: print field and coordinates along trajectories (otherwise $IL = 0$)	0-2	I
XL, YL, B_0	Length; unused; field	cm, unused, kG	3*E
	Entrance face:		
X_E, λ_E, W_E	Integration zone extent; fringe field extent (normally $\simeq$ gap height); wedge angle	cm, cm, rad	3*E
N, C_0–C_5	Unused; fringe field coefficients: $B(s) = B_0 F(s)$, with $F(s) = (1 + \exp(P(s))^{-1}$ 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, tilt (unused if $KPOS$=1) $KPOS = 3$: entrance and exit frames are shifted by YCE and tilted w.r.t. magnet by an angle • either ALE/2 if ALE$\neq$0 • or half the deviation if ALE=0	1-2, 2*cm, rad	I, 3*E



Geometry and parameters in *BEND*: XL = length, B_o = field, θ = deviation.

BINARY	**Binary/Formatted data converter**		
NF	Number of files to convert	≤ 20	I
The next NF lines:			
FNAME	Name of the file to be translated (begin with "B_" iff binary)		A80

BREVOL	**1-D uniform mesh magnetic field map** X-axis cylindrical symmetry is assumed		
IC, *IL*	$IC = 1, 2$: print the map $IL = 1$: print field and coordinates along trajectories	0-2, 0-2	2*I
BNORM	Normalization coefficient: $\frac{\text{desired field}}{\text{field read}}$	no dim.	E
TIT	Title		A80
IX	Number of longitudinal nodes of the map	≤ 400	I
FNAME[1]	Filename (e.g., solenoid.map)		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, 4 or 25	I
XPAS	Integration step	cm	E
KPOS, *XCE*, *YCE*, *ALE*	*KPOS*=1: element aligned, 2: misaligned; shifts, tilt (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* 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
```

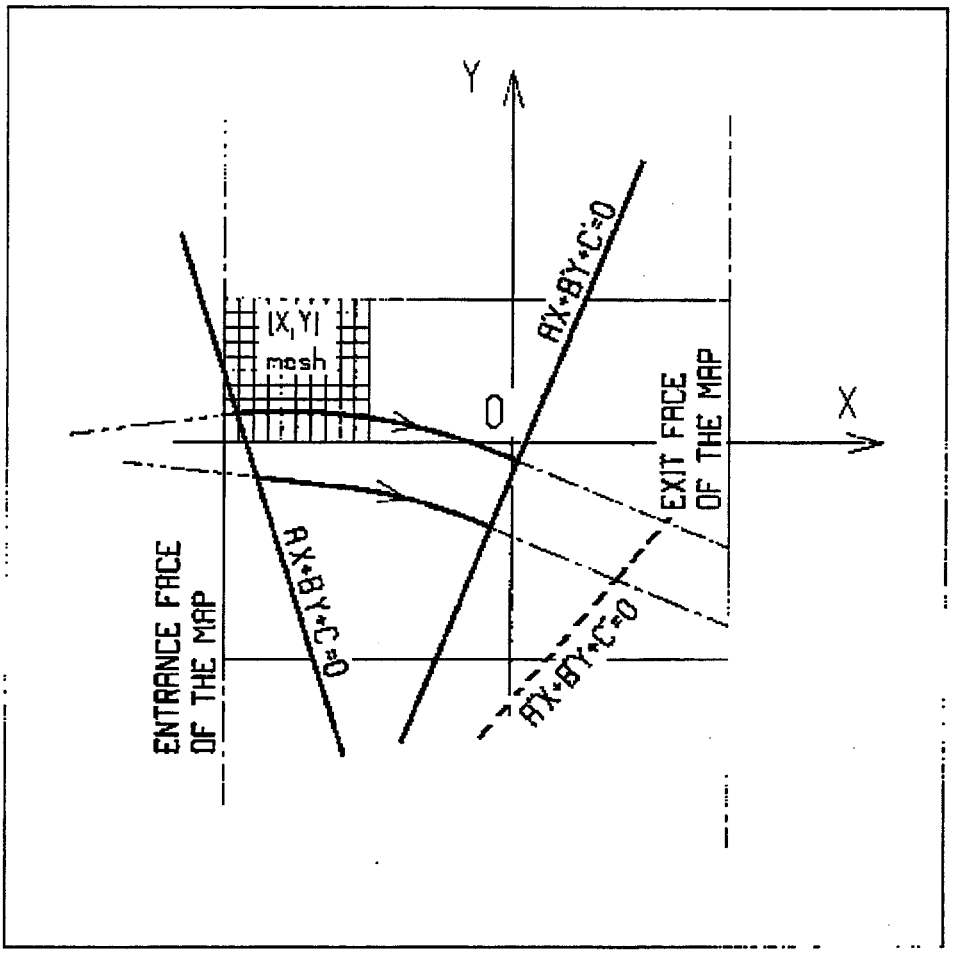
where $X(I)$ and $BX(I)$ are the longitudinal coordinate and field component at node (I) of the mesh. Binary file names *FNAME* must begin with B_. 'Binary' will then automatically be set to '.TRUE.'

CARTEMES	**2-D Cartesian uniform mesh magnetic field map** mid-plane symmetry is assumed		
IC, *IL*	$IC = 1, 2$: print the map $IL = 1, 2$: print field and coordinates along trajectories	0-2, 0-2	2*I
BNORM	Normalization coefficient: $\frac{\text{desired field}}{\text{field read}}$	no dim.	E
TIT	Title		A80
IX, *JY*	Number of longitudinal (IX) and transverse (JY) nodes of the map	≤ 400, ≤ 200	2*I
FNAME[1]	Filename (e.g., spes2.map)		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 stopped 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	Interpolation polynomial order (see *DIPOLE*)	2, 4 or 25	I
XPAS	Integration step	cm	E
KPOS, *XCE*, *YCE*, *ALE*	*KPOS*=1: element aligned, 2: misaligned; shifts, tilt (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* 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_.'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 stopping on the integration boundary $AX + BY + C = 0$, coordinates are extrapolated linearly to the exit face of the map.

CAVITE[1]	**Accelerating cavity** $\Delta W = qV sin(2\pi h f \Delta t + \varphi_s)$		
IOPT	Option	0-3	I
If IOPT=0	Element inactive		
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 $\Delta W_s = q\hat{V} sin\phi_s$		
$\mathcal{L}$, R	Reference closed orbit length; harmonic number	m, no dim.	2*E
$\hat{V}$,ϕ_s	R.F. peak voltage; 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

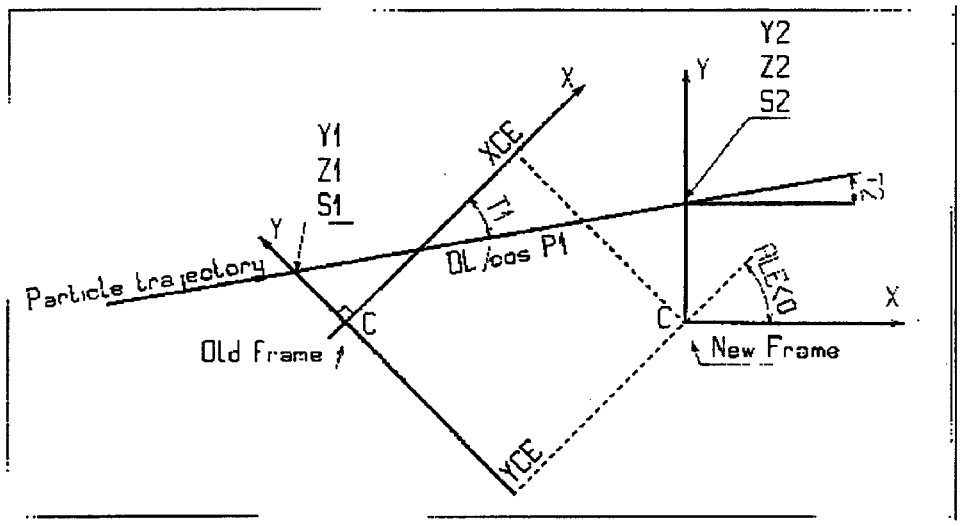
[1] Use *PARTICUL* to declare mass and charge.
[2] For ramping the R.F. frequency following $B\rho(t)$, use *SCALING*, with family *CAVITE*.

CHAMBR	**Long transverse aperture limitation**[1]		
IA	0: element inactive 1: redefinition of the aperture 2: stop testing and reset counters, print information on stopped particles.	0-2	I
$IFORM$, YL[2], ZL, YC, ZC	Taken into account only if $IA = 1$. $IFORM = 1$: rectangular chamber; horizontal (vertical) dimension $\pm YL$ ($\pm ZL$); centered at YC, ZC. $IFORM = 2$: elliptical chamber; horizontal (vertical) axis $\pm YL(\pm ZL)$; centered at YC, ZC.	1-2, 4*cm	I, 4*E

[1]Any particle out of limits is stopped.

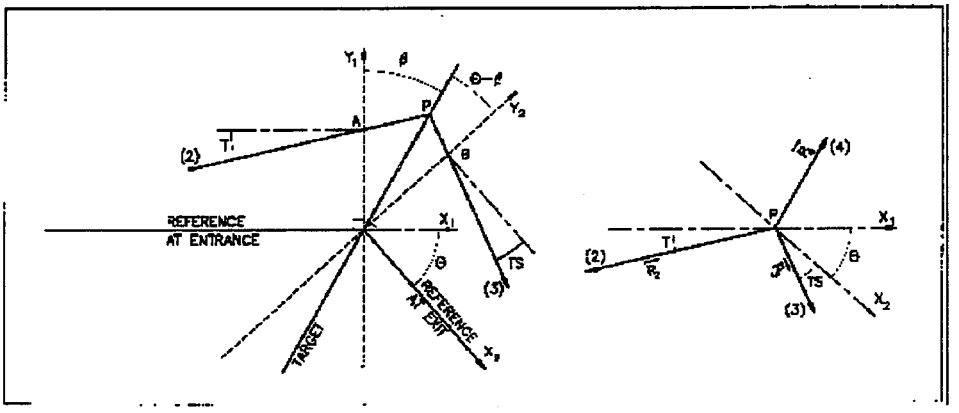
[2]When used with an optical element defined in polar coordinates (e.g. *DIPOLE*) YL is the radius and YC stands for the mean radius (normally, $YC \simeq RM$).

CHANGREF	**Transformation to a new reference frame**		
XCE, YCE, ALE	Longitudinal and transverse shifts, followed by Z-axis rotation	2*cm, deg	3*E



Scheme of the *CHANGREF* procedure.

CIBLE, TARGET	**Generate a secondary beam from target interaction**		
M_1, M_2, M_3, Q T_2, θ, β	Target, incident and scattered particle masses; Q of the reaction; incident particle kinetic energy; scattering angle; angle of the target	5*$\frac{MeV}{c^2}$, 2*deg	7*E
NT, NP	Number of samples in T and P coordinates after *CIBLE*		2*I
TS, PS, DT	Sample step sizes; tilt angle	3*mrad	3*E
BORO	New reference rigidity after *CIBLE*	kG.cm	E



Scheme of the principles of *CIBLE (TARGET)*

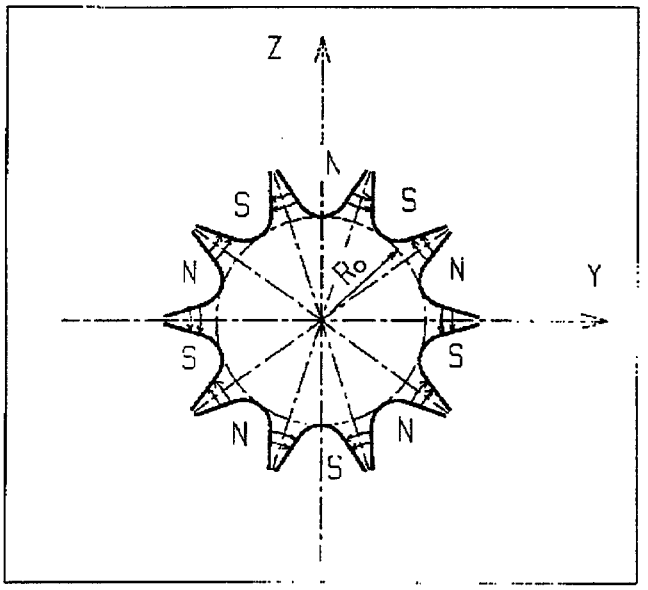
A, T = position, angle of incoming particle 2 in the entrance reference frame
P = position of the interaction
B, T = position, angle of the secondary particle in the exit reference frame
θ = angle between entrance and exit frames
β = tilt angle of the target

CLORB	**Beam Centroid Path; Closed Orbit**		
N	0: inactive ≥ 1: total number of *LABEL*'s at which beam centroid is to be recorded	≥ 0	I
For I = 1, N	A list of N records follows		
LABEL's	N labels at which beam centroid is to be recorded	strings	N*A8

COLLIMA	**Collimator**[1]		
IA	0: element inactive 1: element active 2: element active and print information on stopped particles	0-2	I
IFORM, *YL*, *ZL*, *YC*, *ZC*	Record taken into account only if $IA = 1 - 2$ *IFORM* = 1: rectangular collimator; horizontal (vertical) dimension $\pm YL$ ($\pm ZL$); centered at *YC*, *ZC*. *IFORM* = 2: elliptical collimator; horizontal (vertical) axis $\pm YL$ ($\pm ZL$); centered at *YC*, *ZC*.	1-2, 4*cm	I, 4*E

[1]Any particle out of limits is stopped.

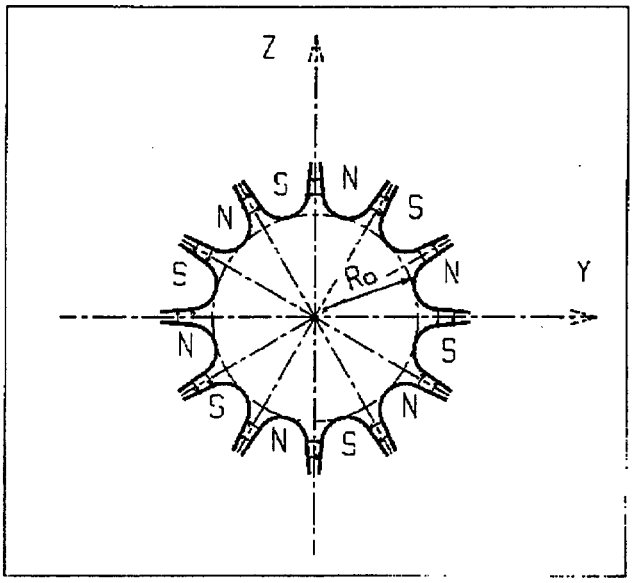
DECAPOLE	**Decapole magnet**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	I
XL, R_0, B_0	Length; radius and field at pole tip	2*cm, kG	3*E
X_E, λ_E	Entrance face: Integration zone extent; fringe field extent ($\lesssim 2R_0$, $\lambda_E = 0$ for sharp edge)	2*cm	2*E
NCE, $C_0 - C_5$	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 (unused if $KPOS$=1)	1-2, 2*cm, rad	I, 3*E



DIPOLE	**Generation of a dipole magnet 2-D map** $B_Z = \mathcal{F} B_0 \left(1 - N\left(\frac{R-RM}{RM}\right) + B\left(\frac{R-RM}{RM}\right)^2 + G\left(\frac{R-RM}{RM}\right)^3\right)$		
NFACE, *IC*, *IL*	Number of field boundaries $IC = 1, 2$: print field map $IL = 1, 2$: print field and coordinates on trajectories	2-3, 0-2, 0-2	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 EFB's; mean radius; minimum and maximum radii	2*deg, 3*cm	5*E
	ENTRANCE FIELD BOUNDARY		
λ, ξ	Fringe field extent (normally equal to gap size); unused 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, 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 $\mid U_{1,2} \mid = \infty$ when $R_{1,2} = \infty$) (Note : $\lambda = 0$, $\omega^+ = ACENT$ and $\theta = 0$ for sharp edge)	2*deg, 4*cm	6*E
	EXIT FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY)		
λ, ξ NC, $C_0 - C_5$, shift	Fringe field parameters	cm, unused 0-6, 6*no dim., cm	2*E 1, 7*E
ω^-, θ, R_1, U_1, U_2, R_2	Positioning and shape of the exit EFB (Note : $\lambda = 0$, $\omega^- = -AT + ACENT$ and $\theta = 0$ for sharp edge)	2*deg, 4*cm	6*E

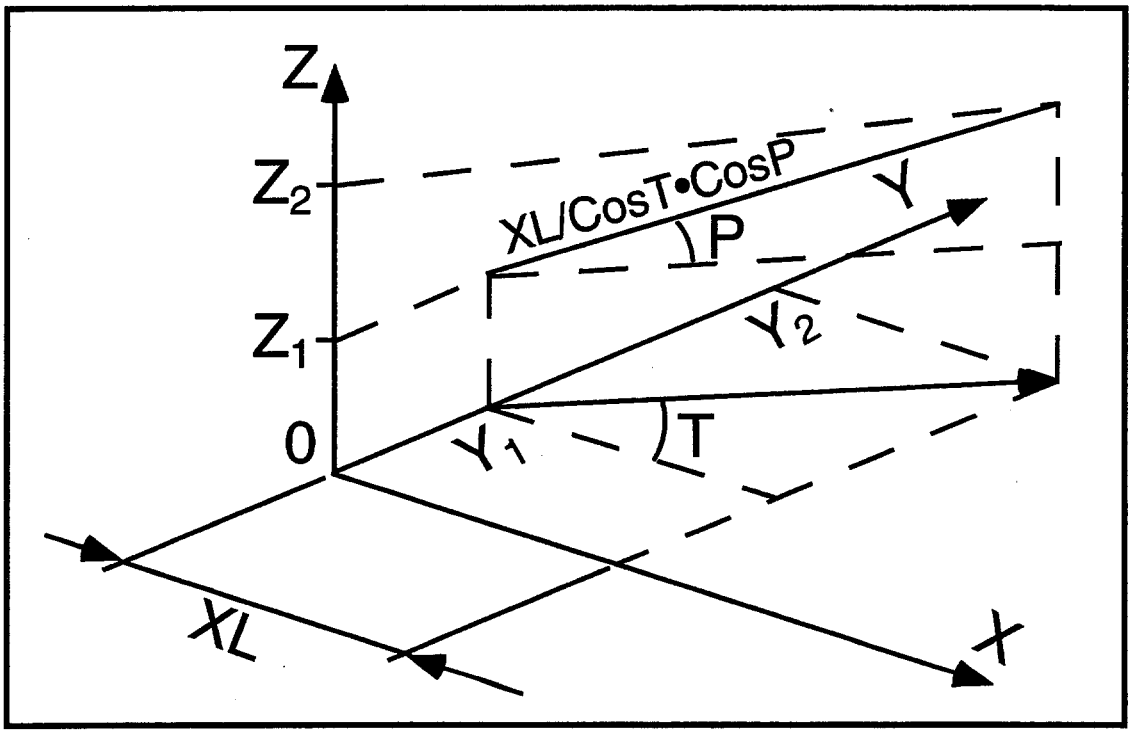
if NFACE = 3	LATERAL FIELD BOUNDARY (See ENTRANCE FIELD BOUNDARY) Next 3 records *only* if *NFACE* = 3		
λ, ξ	Fringe field parameters	cm, unused	2*E
NC, $C_0 - C_5$, shift		0-6, 6* no dim., cm	I,7*E
ω^-, θ, R_1, U_1, U_2, R_2, *RM3*	Positioning and shape of the lateral EFB; *RM3* is the radial position on azimut *ACENT*	2*deg, 5cm	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 $\geq$ 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	Interpolation polynomial order: 2 = second order, 9-point grid 25 = second order, 25-point grid 4 = fourth order, 25-point grid	2, 4 or 25	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

DODECAPO	**Dodecapole magnet**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	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 (unused if $KPOS$=1)	1-2, 2*cm, rad	I, 3*E



DRIFT, ESL **Field-free drift space**

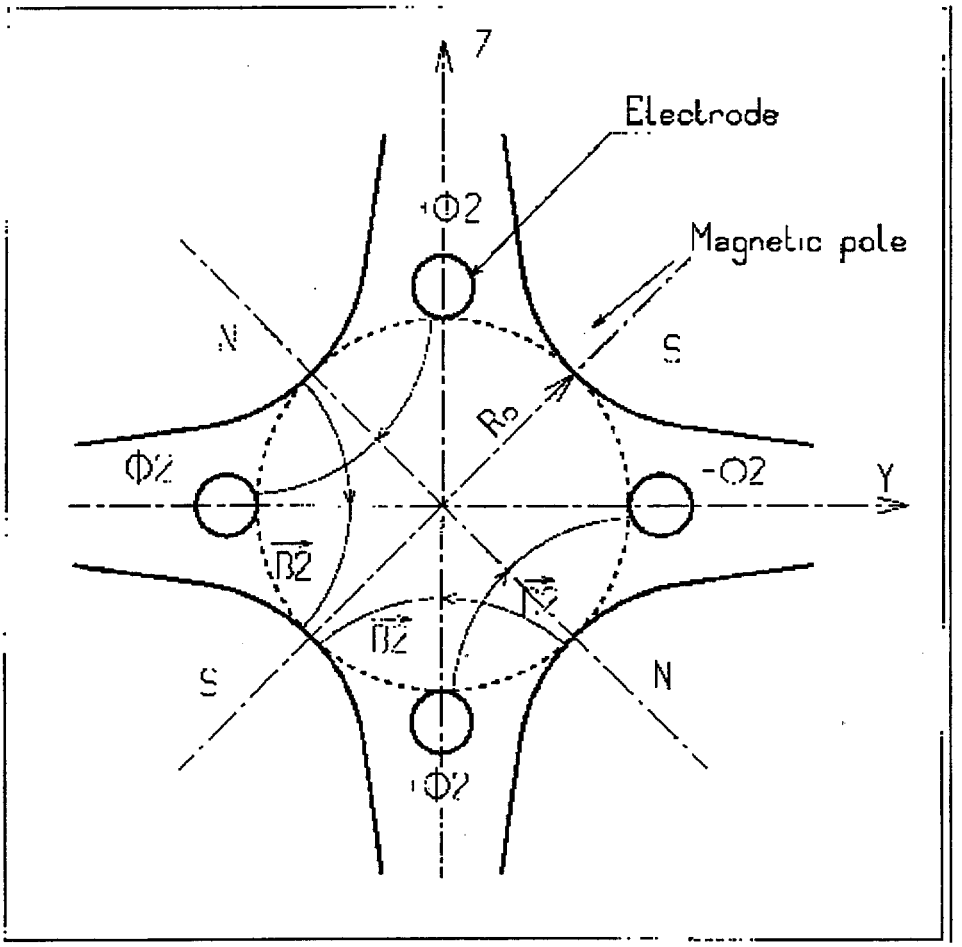
XL	length	cm	E



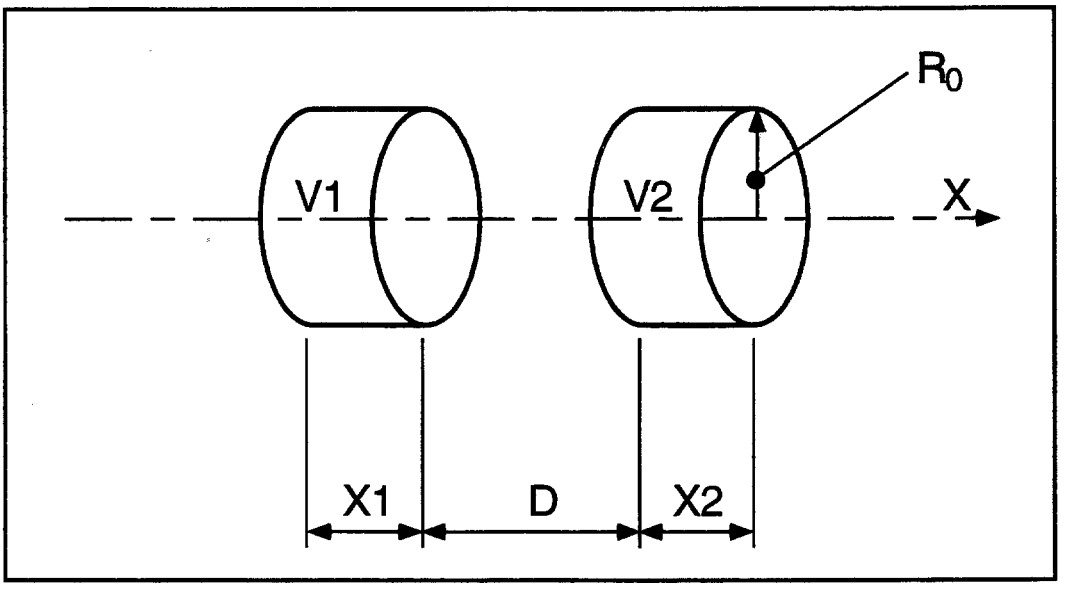
EBMULT[1]	**Electro-magnetic Multipole**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	I
	Electric poles		
XL, R_0, E_1, E_2, E_3, ..., E_{10}	Length of element; radius at pole tip; field at pole tip for dipole, quadrupole, sextupole, octupole, decapole and dodecapole electric components	2*cm, 10*kG	12*E
	Entrance face		
X_E, λ_E, E_2, E_3, E_4, E_5, E_6	Integration zone; fringe field extent; dipole fringe field extent = λ_E; quadrupole fringe field extent = $\lambda_E * E_2$; sextupole fringe field extent = $\lambda_E * E_3$; octupole fringe field extent = $\lambda_E * E_4$; decapole fringe field extent = $\lambda_E * E_5$; dodecapole fringe field extent = $\lambda_E * E_6$ (for any component: sharp edge if field extent is zero)	2*cm, 5*no dim.	7*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, S_4, S_5, S_6	Integration zone; as for entrance	2*cm, 4*no dim.	7*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
R_1, R_2, R_3, ..., R_{10}	Skew angles of electric field components	10*rad	10*E
	Magnetic poles		
XL, R_0, B_1, B_2, B_3, ..., B_{10}	Length of element; radius at pole tip; field at pole tip for dipole, quadrupole, sextupole, octupole, decapole and dodecapole magnetic components	2*cm, 10*kG	12*E
	Entrance face		
X_E, λ_E, E_2, E_3, E_4, E_5, E_6	Integration zone; fringe field extent; dipole fringe field extent = λ_E; quadrupole fringe field extent = $\lambda_E * E_2$; sextupole fringe field extent = $\lambda_E * E_3$; octupole fringe field extent = $\lambda_E * E_4$; decapole fringe field extent = $\lambda_E * E_5$; dodecapole fringe field extent = $\lambda_E * E_6$ (for any component: sharp edge if field extent is zero)	2*cm, 5*no dim.	7*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_3, S_4, S_5, S_6	Integration zone; as for entrance	2*cm, 4*no dim.	7*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
R_1, R_2, R_3, ..., R_{10}	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 (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E

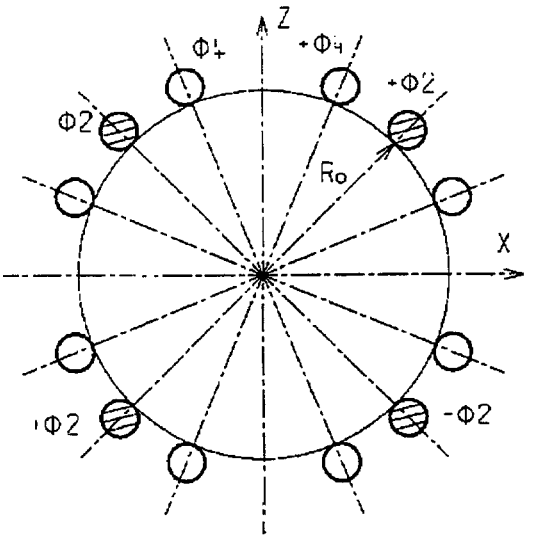


EL2TUB[1]	**Two-tubes electrostatic lens**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	I
X_1, D, X_2, R_0	Length of first tube; distance between tubes; length of second tube; radius	3*m	4*E
V_1, V_2	Potentials	2*V	2*E
XPAS	Integration step	cm	E
KPOS, *XCE*, *YCE*, *ALE*	*KPOS*=1: element aligned, 2: misaligned; shifts, tilt (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E



[1]Use *PARTICUL* to declare mass and charge.

ELMULT[1]	**Electric Multipole**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	I
XL, R_0, E_1, E_2, E_3, ..., E_{10}	Length of element; radius at pole tip; field at pole tip for dipole, quadrupole, sextupole, octupole, decapole and dodecapole components	2*cm, 10*kG	12*E
	Entrance face		
X_E, λ_E, E_2, E_3, E_4, E_5, E_6	Integration zone; fringe field extent; dipole fringe field extent = λ_E; quadrupole fringe field extent = $\lambda_E * E_2$; sextupole fringe field extent = $\lambda_E * E_3$; octupole fringe field extent = $\lambda_E * E_4$; decapole fringe field extent = $\lambda_E * E_5$; dodecapole fringe field extent = $\lambda_E * E_6$ (sharp edge if field extent is zero)	2*cm, 5*no dim.	7*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, S_4, S_5, S_6	Integration zone; as for entrance	2*cm, 4*no dim.	7*E
NCS, $C_0 - C_5$		0-6, 6*no dim.	I, 6*E
R_1, R_2, R_3, ..., R_{10}	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 (unused if $KPOS$=1)	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 $IL = 1, 2$: print field and coordinates along trajectories	0-2, 0-2	2*I
ENORM	Normalization coefficient: $\frac{\text{desired field}}{\text{field read}}$	no dim.	E
TIT	Title		A80
IX	Number of longitudinal nodes of the map	≤ 400	I
FNAME[2]	Filename (e.g., elens.map)		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, 4 or 25	I
XPAS	Integration step	cm	E
KPOS, XCE, *YCE, ALE*	*KPOS*=1: element aligned, 2: misaligned; shifts, tilt (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E

[1]Use *PARTICUL* to declare mass and charge.

[2]*FNAME* 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_. 'Binary' will then automatically be set to '.TRUE.'

FAISCEAU **Print particle coordinates**

Print particle coordinates at the location where the keyword is introduced in the structure.

FAISCNL **Store particle coordinates in file FNAME**

FNAME[1,2]	Name of storage file (e.g. zgoubi.fai).	A80

FAISCNLA **Store coordinates every *IP* other pass [,at labeled elements]**

FNAME[1] [, *LABEL(s)*]	Name of storage file (e.g. zgoubi.fai) [; 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

[1] *FNAME* contains the particle coordinates and other informations. These are stored following the *FORTRAN* sequence below.

```
OPEN (UNIT = NL, FILE = FNAME)
DO 1 I = 1, IMAX
 WRITE (NL, 100) LET(I),IEX(I),(FO(J,I), J=1,6),(F(J,I),J=1,6),DUM,IDUM,I,IREP(I),
    ¿  SORT(I),DUM,DUM,DUM,DUM,PH(I),DP(I),DUM,DUM,DUM,BORO,IPASS,DMY,HMS,KLEY,LABEL,NOEL-1
100     FORMAT (1X, A1, 1X, I2, IP, 6E16.8,
                /, 3E24.16,
                /, 3E24.16, E16.8,
                /, I1, 2I3, 7E16.8,
                /, 7E16.8, I1, 2I3,
                /, 7E16.8,
                /, 4E16.8, I6, 1X, A9, 1X, A8, 1X, A8, 1X, A10, I5)
 1    CONTINUE
```

The signification of these parameters is given as follows

DUM, IDUM: dummies
From LET(I) to IREP(I): see *OBJET*
SORT(I): path length of particles stopped by walls (*CHAMBR, COLLIMA*)
PH(I), DP(I): phase and momentum kick at traversal of *CAVITE*
BORO: reference rigidity (see *OBJET*)
DMY, HMS: date, time
KLEY, LABEL, NOEL-1: last encountered keyword, its *LABEL* and position

[2] These data can in turn be read directly from the file *FNAME* by means of *OBJET*, *KOBJ* = 3.

FIN, END **End of input data list**

Any information following these keywords will be ignored

FIT	**Fitting procedure**		
NV	Number of physical parameters to be varied	≤ 20	I
For I = 1, NV	repeat NV times the following sequence		
IR, *IP*, *XC*, *DV*	Number of the element in the structure; number of the physical parameter in the element; coupling switch (off = 0); allowed ± range of variation of the parameter.	≤ 200, ≤ 99, $\pm$ 200.99, relative	2*I, 2*E
NC	Number of constraints	≤ 20	I
For I = 1, NC	repeat NC times the following sequence		
IC, *I*, *J*, *IR*, V[1], *WV*	*IC*, *I* and *J* define the type of constraint (see table below); number of the element at the exit of which the constraint applies; value; weight of the constraint (the lower the stronger).	0-3, 1-200, current unit[1], no dim.	4*I, 2*E

Type of constraint	**Parameters defining the constraint**			
	IC	**I**	**J**	**Constraint**
Beam matrix[2]	0	1 - 4	1 - 4	σ_{IJ}
First order transfer coefficients[2]		1 - 6	1 - 6	R_{IJ}
	1	7	any	Horizontal determinant
		8	any	Vertical determinant
Second order transfer coefficients[3]	2	1 - 6	11 - 66	$T_{I,j,k}$ $(j = [J/10], k = J - 10[J/10])$
Trajectory coordinate[4]	3	1 - *IMAX*	1 - 6[5]	$F(J, I)$

[1] The unit of V is that specified in the corresponding keyword.
[2] It is advised to use *OBJET* and *KOBJ* = 5, for the definition of the initial coordinates.
[3] It is advised to use *OBJET* and *KOBJ* = 6, for the definition of the initial coordinates.
[4] For use normally with object definition by *OBJET*. Thus, I = trajectory number = 1 to *IMAX* if *KOBJ* $\neq 2$; I = trajectory number = 1 to 7 if *KOBJ* = 2.
[5] J = coordinate number = 1 to 6 for respectively D, Y, T, Z, P or X.

FOCALE	**Particle coordinates and horizontal beam dimension at distance XL**		
XL	Distance from the position of the keyword	cm	E

FOCALEZ	**Particle coordinates and vertical beam dimension at distance XL**		
XL	Distance from the position of the keyword	cm	E

GASCAT	**Gas Scattering**		
KGA	Off/On switch	0, 1	I
AI, *DEN*	Atomic number; density		2*E

HISTO	**1-D histogram**		
J, $X_{\min}$, $X_{\max}$, NBK, NH	J = type of coordinate to be histogramed; 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 independency 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 histogramed, 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**

MAP2D	**2-D Cartesian uniform mesh magnetic field map** no symmetry		
IC, *IL*	$IC = 1, 2$: print the field map $IL = 1, 2$: print field and coordinates along trajectories	0-2, 0-2	2*I
BNORM	Normalization coefficient: $\frac{\text{desired field}}{\text{field read}}$	no dim.	E
TIT	Title		A80
IX, *JY*	Number of longitudinal and transverse nodes of the mesh	$\leq 400, \leq 200$	2*I
FNAME[1]	File name (e.g., magnet.map)		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	Interpolation polynomial order See *DIPOLE*	2, 25	I
XPAS	Integration step	cm	E
KPOS, *XCE*, *YCE*, *ALE*	*KPOS*=1: element aligned, 2: misaligned; shifts, tilt (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* contains the field map data. These must be formatted according to the following *FORTRAN* read sequence (that normally fits *TOSCA* code *OUTPUTS*):

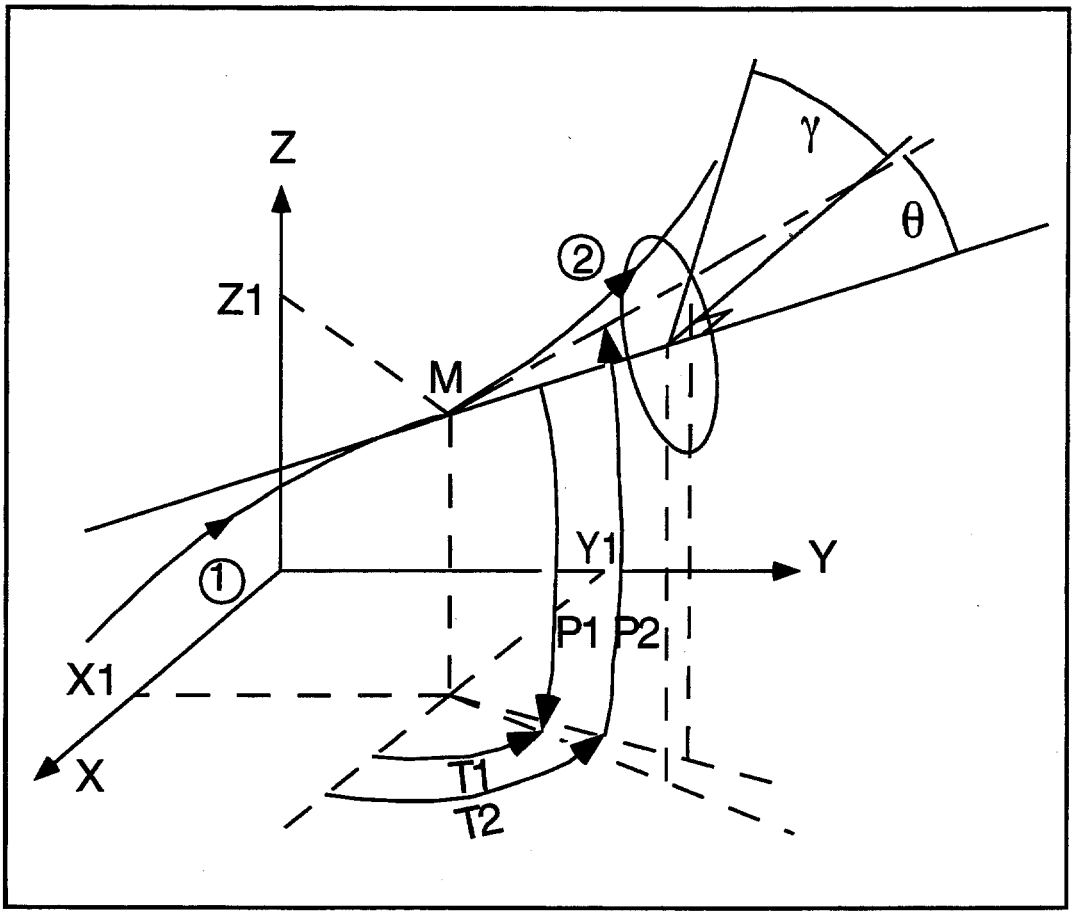
```
      OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD')
      DO 1 J = 1, JY
        DO 1 I = 1, IX
          IF (BINARY) THEN
                READ(NL) Y(J), X(I), BY(I,J), BZ(I,J), BX(I,J)
          ELSE
                READ(NL,100) Y(J), X(I), BY(I,J), BZ(I,J), BX(I,J)
  100           FORMAT (1X, 6E11.2)
              ENDIF
    1     CONTINUE
```

where $X(I)$, $Y(J)$, Z, are the longitudinal, horizontal and vertical coordinates, and BX, BY, BZ are the components of the field at a node (I, J, K) of the map.

For binary files, *FNAME* must begin with B_; 'Binary' will then automatically be set to '.TRUE.'

MATRIX	**Calculation of transfer coefficients**		
IORD, IFOC	Options : *IORD* = 0: Same effect as *FAISCEAU* 1: First order transfer matrix 2: First order transfer matrix R_{ij}, second order array T_{ijk}, and higher order transfer coefficients *IFOC* = 0: matrix at actual position, reference ≡ particle # 1 1: matrix at the closest first order horizontal focus, reference ≡ particle # 1 10 + NPER: same as *IFOC* = 0, and also calculates the twiss parameters and tune numbers (assuming that the DATA file describes one period of a NPER-period structure).	0-2, 0-1 or > 10	2*I

MCDESINT[1]	**Monte-Carlo simulation of in-flight decay** M1 → M2 + M3		
M2, *M3*	Masses of the two decay products	2*MeV/c^2	2*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 abandoned. 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.

MCOBJET	**Monte Carlo generation of a 3-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	≤ 200	I
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	**Generation in a window**		
δY, δT, δZ, δP, δX, δD	Distribution widths, depending on KY, KT etc.[1]	m, rad, m, rad, m, no dim.	5*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) (unused if $KD = 1$)	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, $p(x) = 1$ 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, $p(D) = 1$ 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 = 2	**Generation 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
$\delta Y, \delta T, \delta Z, \delta P,$ $\delta X, \delta D$	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 $P(D)$ (unused if $KOBJ = 3$)	no dim.	5*E
$IR1, IR2, IR3$	Random sequence seeds	3*$\simeq 10^6$	3*I
if KOBJ = 3	**Generation in a phase-space ellipse**[1]		
$\alpha_Y, \beta_Y, \varepsilon_Y/\pi, N_{\sigma_{\epsilon_Y}}$ [, $N'_{\sigma_{\epsilon_Y}}$ if $N_{\sigma_{\epsilon_Y}} < 0$][2]	Ellipse parameters and normalized emittance, Y-T phase-space; cut-off	no dim., m/rad, m.rad, units of $\sigma(\varepsilon_Y)$	3*E, I
$\alpha_Z, \beta_Z, \varepsilon_Z/\pi, N_{\sigma_{\epsilon_Z}}$ [, $N'_{\sigma_{\epsilon_Z}}$ if $N_{\sigma_{\epsilon_Z}} < 0$][2]	Ellipse parameters and normalized emittance, Z-P phase-space; cut-off	no dim., m/rad, m.rad, units of $\sigma(\varepsilon_Z)$	3*E, I
$\alpha_X, \beta_X, \varepsilon_X/\pi, N_{\sigma_{\epsilon_X}}$ [, $N'_{\sigma_{\epsilon_X}}$ if $N_{\sigma_{\epsilon_X}} < 0$][2]	Ellipse parameters and normalized emittance, X-D phase-space; cut-off	no dim., m/rad, m.rad, units of $\sigma(\varepsilon_X)$	3*E, I [,I]
$IR1, IR2, IR3$	Random sequence seeds	3*$\simeq 10^6$	3*I

[1] Similar possibilities, non-random, are offered with *OBJET*, KOBJ=8

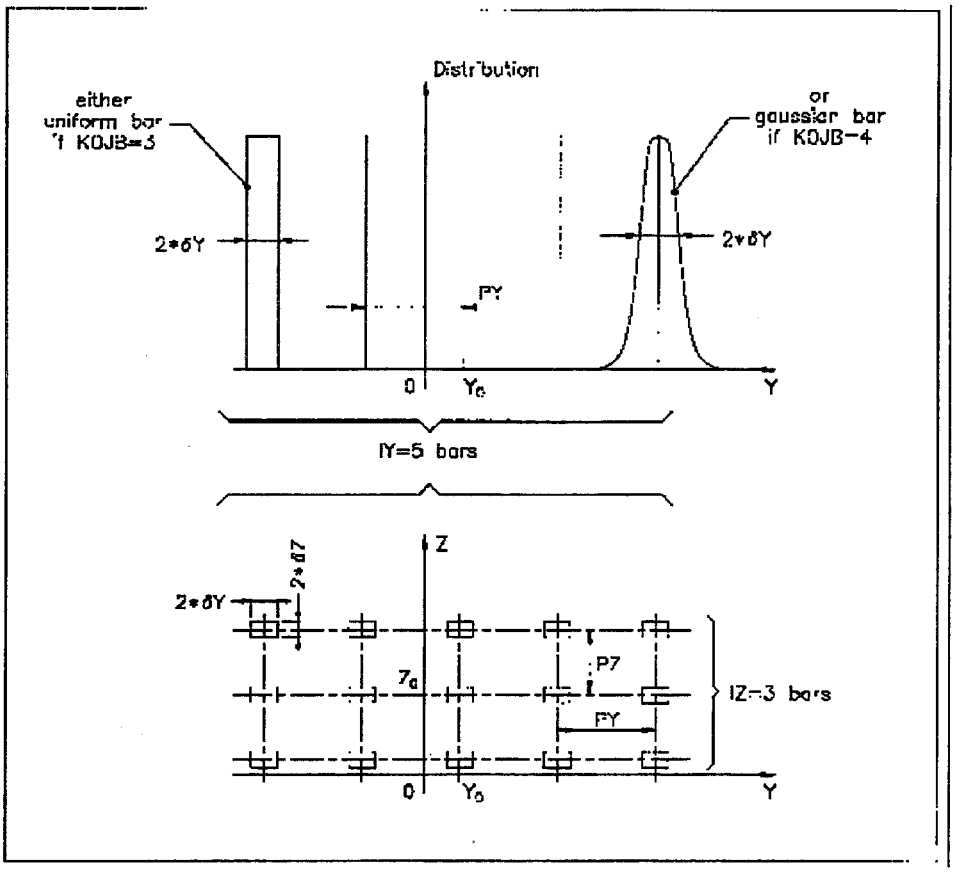
[2] Sorting within the ellipse frontier

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

if $N_{\sigma_{\epsilon_Y}} > 0$, or within the ring

$$[\,|N_{\sigma_{\epsilon_Y}}|, N'_{\sigma_{\epsilon_Y}}\,]$$

if $N_{\sigma_{\epsilon_Y}} < 0$.



Scheme of the input parameters to *MCOBJET* when $KOBJ = 3, 4$

A: A distribution of the Y coordinate
B: 2-D grid in (Y, Z) space.

MULTIPOL	**Magnetic Multipole**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	I
XL, R_0, B_1, B_2, ..., B_{10},	Length of element; radius at pole tip; field at pole tip for dipole, quadrupole, sextupole, octupole, decapole and dodecapole components	2*cm,10*kG	12*E
	Entrance face		
X_E, λ_E, E_2, E_3, E_4, E_5, E_6	Integration zone; fringe field extent; dipole fringe field extent = λ_E; quadrupole fringe field extent = $\lambda_E * E_2$; sextupole fringe field extent = $\lambda_E * E_3$; octupole fringe field extent = $\lambda_E * E_4$; decapole fringe field extent = $\lambda_E * E_5$; dodecapole fringe field extent = $\lambda_E * E_6$ (sharp edge if field extent is zero)	2*cm,5*no dim.	7*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, S_4, S_5, S_6	Integration zone; as for entrance	2*cm, 4*no dim.	7*E
NCS, $C_0 - C_5$		0-6, *no dim.	I, 6*E
R_1, R_2, R_3, ..., R_{10}	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 (unused if $KPOS$=1) for *QUADRUPO*. $KPOS$ = 3: effective only if $B_1 \neq 0$: entrance and exit frames are shifted by YCE and tilted w.r.t. magnet by an angle • either ALE/2 if ALE≠0 • or half the deviation if ALE=0	1-2, 2*cm, rad	I, 3*E

OBJET	**Generation of an object**		
BORO	Reference rigidity	kG.cm	E
KOBJ	Option index	1-6	I
if KOBJ = 1	Generation of a symmetric object		
IY, *IT*, *IZ*, *IP*, *IX*, *ID*	Ray-Tracing assumes mid-plane symmetry Total number of points in $\pm Y$, $\pm T$, $\pm Z$, $\pm P$, $\pm X$ and $\pm D$ coordinates ($IY \leq 20$,...,$ID \leq 20$)	IY*IT*IZ*IP* *IX*ID ≤ 200	6*1
PY, *PT*, *PZ*, *PP*, *PX*, *PD*	Step size in Y, T, Z, P, X and momentum ($PD = \delta B\rho/BORO$)	cm, mrad, cm, mrad, cm, no dim.	6*E E
$\mathcal{D}$	Reference relative momentum $B\rho/BORO$	no dim.	
if KOBJ = 2	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)	$IMAX \leq 200$	2*I
For I = 1, IMAX	Repeat *IMAX* times the following line		
Y, *T*, *Z*, *P*, *X*, *D*, *LET*	Coordinates and tagging character of the *IMAX* particles ($D = B\rho/BORO$)	cm, mrad, cm, mrad, cm, no dim., char.	6*E, A1
$IEX(I = 1, IMAX)$	*IMAX* times 1 or -2. If $IEX(I) = 1$, trajectory number I is calculated. If $IEX(I) = -2$, it is not calculated	1 or -2	*IMAX*I
If KOBJ=3	Reads coordinates from a storage file		
IMAX, IDMAX	Total number of particles and distinct momenta (For more than 200 particles stored in *FNAME*, use '*REBELOTE*')	≤ 200, ≤ 20	2*I
FNAME[1]	File name (e.g., zgoubi.rays)		A80

[1] *FNAME* contains the particle coordinates. These must be formatted according to the following *FORTRAN* sequence

```
OPEN (UNIT = NL, FILE = FNAME, STATUS = 'OLD')
DO 1 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)
 100   FORMAT (1X, A1, 1X, I2, 6E16.8, / , 6E16.8, 2I3 / )
 1   CONTINUE
```

with the following meaning for output variables:

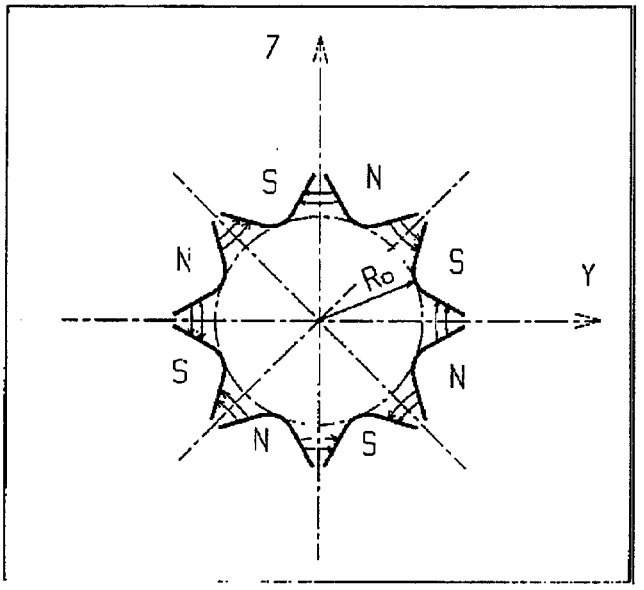
LET: tagging letter ; IEX: flag ; FO(1-6,I), initial coordinates of particle number I: relative momentum, horizontal and vertical coordinates, path length ; F(1-6,I), current coordinates of particle number I; IREP:flag.

If KOBJ = 4	**Generation of a non symmetric object**		
$IY, IT, IZ, IP,$ IX, ID	Total number of points in $\pm Y$, $\pm T$, $+Z$, $+P$, $\pm X$ and $\pm D$ coordinates ($IY \leq 20$,...,$ID \leq 20$)	IY*IT*IZ*IP *IX*ID ≤ 200	6*I
$PY, PT, PZ, PP,$ PX, PD	Step sizes in Y, T, Z, P X and D.	cm, mrad, cm, mrad, cm, no dim.	6 *E
$\mathcal{D}$	Reference relative momentum $B\rho/BORO$	no dim.	E
If KOBJ = 5	**Generation of 11 particles,** for the calculation of first order transfer coefficients with *MATRIX*		
$PY, PT, PZ, PP,$ PX, PD	Step sizes in Y, T, Z, P, X and D	cm, mrad, cm, mrad, cm, no dim.	6*E
$YR, TR, ZR, PR,$ XR, DR	Reference trajectory; $DR = B\rho/BORO$	cm, mrad, cm, mrad, cm, no dim.	6*E
If KOBJ = 6	**Generation of 61 particles** for the calculation of first and higher order transfer coefficients with *MATRIX*		
$PY, PT, PZ, PP,$ PX, PD	Step sizes in Y, T, Z, P, X and D cm, no dim.	cm, mrad, cm, mrad	6*E
$YR, TR, ZR, PR,$ XR, DR	Reference trajectory; $DR = B\rho/BORO$	cm, mrad, cm, mrad, cm, no dim.	6*E
If KOBJ = 7	**Object with kinematics**		
$IY, IT, IZ, IP,$ IX, ID	Number of points in $\pm Y$, $\pm T$,$\pm Z$, $\pm P$, $\pm X$; ID is not used	IY*IT*IZ*IZ*IX* $IP \leq 200$	6*I
$PY, PT, PZ, PP,$ PX, PD	Step sizes in Y, T, Z, P and X; PD = kinematic coefficient, such that $D(T) = \mathcal{D} + PD * T$	cm, mrad, cm, mrad, cm, mrad^{-1}	6*E
$\mathcal{D}$	$B\rho/BORO$	no dim.	E
If KOBJ = 8	**Generation of phase-space coordinates on a 6-D ellipsoïd**[1]		
IY, IZ, IX	Number of samples in each 2-D phase-space		3*I
$Y_0, T_0, Z_0, P_0,$ X_0, D_0	Central values ($D_0 = B\rho/BORO$)	m, rad, m, rad, m, no dim.	6*E
$\alpha_Y, \beta_Y, \varepsilon_Y/\pi$ $\alpha_Z, \beta_Z, \varepsilon_Z/\pi$ $\alpha_X, \beta_X, \varepsilon_X/\pi$	ellipse parameters and emittances (see *MCOBJET*, KOBJ=3)	no dim., m/rad, m.rad	3*E

[1]Similar possibilities, random, are offered with *MCOBJET*, KOBJ=3

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)	≤ 200	I
$M_1 - M_5$	Rest masses of the bodies	5*GeV/c^2	5*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	cm, mrad, cm, mrad, no dim.	5*E
δY, δT, δZ, δP, δD		cm, mrad, 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	$IL = 1, 2$: print field and coordinates along trajectories	0-2	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 (unused if $KPOS$=1)	1-2, 2*cm, rad	I, 3*E



Octupole magnet

ORDRE **Higher order Taylor expansions in lenss**

IO	$IO = 4$: expansions of $\vec{R}$ and $\vec{u}$ up to $\vec{u}^{(4)}$ (default option)	4 or 5	I
	$IO = 5$: expansions of $\vec{R}$ and $\vec{u}$ up to $\vec{u}^{(5)}$		
	This option applies to *QUADRUPO*, *SEXTUPOL*, *OCTUPOLE*, *DECAPOLE*, *DODECAPO*, *MULTIPOL*, *EBMULT* and *ELMULT*.		

PARTICUL **Particle Characteristics**

M, Q, G, τ, X	Mass; charge; gyromagnetic factor; COM life-time; unusued	MeV/c^2, C, no dim., s 5*E

NOTE : Only the parameters of concern need their value be specified (for instance M, Q for electric lenss); others can be set to zero.

PLOTDATA Coordinate Output for PLOTDATA Graphic Software [25]

To be documented.

POISSON	**Read field data from POISSON output**		
IC, *IL*	$IC = 1, 2$: print the field map $IL = 1, 2$: print field and coordinates along trajectories	0-2, 0-2	2*I
BNORM	Normalization coefficient: $\frac{\text{desired field}}{\text{field read}}$	no dim.	E
TIT	Title		A80
IX, *IY*	Number of longitudinal and transverse nodes of the uniform mesh	≤ 400, ≤ 200	2*I
FNAME[1]	Filename (normally, outpoi.lis)		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	Interpolation polynomial order as for *DIPOLE*	2, 4 or 25	I
XPAS	Integration step	cm	E
KPOS, *XCE*, *YCE*, *ALE*	*KPOS*=1: element aligned, 2: misaligned; shifts, tilt (unused if *KPOS*=1)	1-2, 2*cm, rad	I, 3*E

[1] *FNAME* contains the field map data. These must be formatted according to the following *FORTRAN* read sequence:

```
    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.

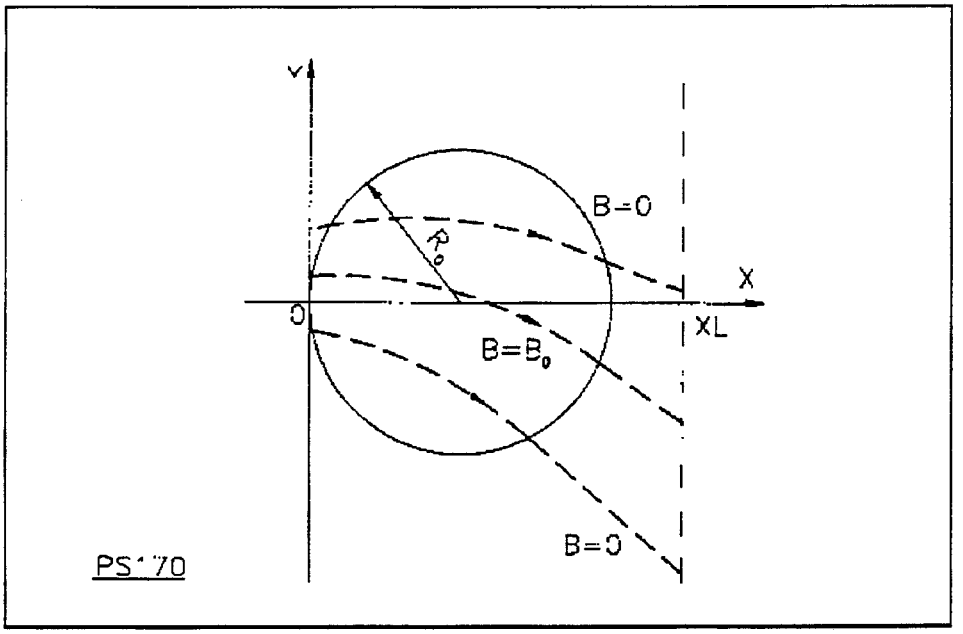
POLARMES	**2-D polar mesh magnetic field map** mid-plane symmetry is assumed		
IC, *IL*	*IC* = 1, 2: print the map *IL* = 1, 2: print field and coordinates along trajectories	0-2, 0-2	2*I
BNORM	Normalization coefficient: $\frac{\text{desired field}}{\text{field read}}$	no dim.	E
TIT	Title		A80
IX, *JY*	Number of angular (*IX*) and radial (*JY*) nodes of the map	≤ 400, ≤ 200	2*I
FNAME[1]	Filename (e.g., spes2.map)		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	Interpolation polynomial order (see *DIPOLE*)	2, 4 or 25	I
XPAS	Integration step	cm	E
KPOS	as for *DIPOLE*. 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* 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 F8.1)
      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_ .'Binary' will then automatically be set to '.TRUE.'

PS170	**Simulation of a round shape dipole**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	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 (unused if *KPOS*=1)	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	$IL = 1, 2$: print field and coordinates along trajectories	0-2	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 (unused if $KPOS$=1)	1-2, 2*cm, rad	I, 3*E

QUADRUPO	**Quadrupole Magnet**		
IL	$IL = 1, 2$: print field and coordinates along trajectories	0-2	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 (unused if $KPOS$=1)	1-2, 2*cm, rad	I, 3*E