Copyright Notice for TRIVAC

The development of TRIVAC is financially supported, directly or indirectly, by various organizations including École Polytechnique de Montréal, Hydro-Québec and the Hydro-Québec chair in nuclear engineering, the Natural Science and Engineering Research Council of Canada (NSERC), Atomic Energy of Canada limited (AECL) and the CANDU Owners Group (COG). The code TRIVAC and its users guide are and will remain the property of École Polytechnique de Montréal.

Trivac is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

Permission is granted to the public to copy TRIVAC without charge. École Polytechnique de Montréal, makes no warranty, express or implied, and assumes no liability or responsibility for the use of TRIVAC.
Acknowledgments

The computer code TRIVAC results from a concerted effort made at École Polytechnique de Montréal. The main authors of this report would therefore like to express their thanks to École Polytechnique de Montréal for its support along the years as well as to the graduate students and research associates which have contributed to the development of TRIVAC along the years. Finally, the TRIVAC team would never have survived without the financial support of the Natural Science and Engineering Research Council of Canada (NSERC), Hydro-Québec and Atomic Energy of Canada Limited (AECL).
SUMMARY

TRIVAC is a computer code intended to compute the neutron flux in a fractional or in a full core representation of a nuclear reactor. Interested readers can obtain fundamental informations about full-core calculations in Chapter 5 of Ref. 1. The multigroup and multidimensional form of the diffusion equation or simplified $P_n$ equation is first discretized to produce a consistent matrix system. This matrix system is subsequently solved using iterative techniques (inverse or preconditioned power method with ADI preconditioning) and sparse matrix algebra techniques (triangular factorization). The actual implementation of TRIVAC allows the discretization of 1-D geometries (slab and cylindrical), 2-D geometries (Cartesian, cylindrical and hexagonal) and 3-D geometries (Cartesian and hexagonal). Many discretization techniques are available, including mesh corner or mesh centered finite difference methods, collocation techniques of various order and finite element methods based on a primal or dual functional formulation. TRIVAC also permits the equations of the generalized perturbation theory (GPT) to be solved as fixed source eigenvalue problems. Finally, several implicit numerical schemes are available for the solving of space-time neutron kinetics problems.

The execution of TRIVAC is controlled by the generalized GAN driver. It is modular and can be interfaced easily with other production codes.
## Contents

- Copyright Notice for TRIVAC .................................................. ii
- Acknowledgments .................................................................. iii
- Contents .................................................................................. v
- List of Figures .......................................................................... vi
- List of Tables ............................................................................ vii

### 1 INPUT DATA SPECIFICATIONS .................................................. 1
  1.1 Syntactic rules for input data specifications ................................. 1
  1.2 The global input structure ....................................................... 1
  1.3 The GEO: module .................................................................. 2
    1.3.1 Data input for module GEO: ........................................... 3
    1.3.2 Examples of geometries .................................................. 12
  1.4 The MAC: module .................................................................. 13
    1.4.1 Data input for module MAC: ........................................... 13
    1.4.2 Description of the nuclear data ........................................ 15
  1.5 The BIVACT: module ............................................................ 18
    1.5.1 Data input for module BIVACT: ....................................... 18
  1.6 The TRIVAT: module ............................................................. 21
    1.6.1 Data input for module TRIVAT: ....................................... 21
  1.7 The BIVACA: module ........................................................... 24
    1.7.1 Data input for module BIVACA: ...................................... 24
  1.8 The TRIVAA: module ............................................................ 25
    1.8.1 Data input for module TRIVAA: ....................................... 25
  1.9 The FLUD: module ............................................................... 27
    1.9.1 Data input for module FLUD: .......................................... 27
  1.10 The DELTA: module ............................................................ 30
    1.10.1 Data input for module DELTA: ....................................... 30
  1.11 The GPTFLU: module .......................................................... 32
    1.11.1 Data input for module GPTFLU: ...................................... 32
  1.12 The OUT: module ............................................................... 35
    1.12.1 Data input for module OUT: .......................................... 35
  1.13 The ERROR: module ............................................................ 37
  1.14 The INIKIN: module ............................................................ 39
    1.14.1 Data input for module INIKIN: ....................................... 39
  1.15 The KINSOL: module ........................................................... 41
    1.15.1 The direct (forward) solution ......................................... 41
    1.15.2 The adjoint (backward) solution ..................................... 42
    1.15.3 The calling specifications .............................................. 43
    1.15.4 Data input for module KINSOL: ...................................... 44
  1.16 The VAL: module ............................................................... 46
    1.16.1 Data input for module VAL: .......................................... 46

### 2 EXAMPLES OF INPUT DATA FILES ........................................... 48
  2.1 IAEA-2D benchmark ............................................................. 48
  2.2 Biblis-2D benchmark ........................................................... 49
  2.3 IAEA-3D benchmark ............................................................ 52
  2.4 S30 hexagonal benchmark in 2-D ........................................... 55
  2.5 LMW benchmark in 2-D ........................................................ 56

- References ............................................................................... 60
- Index ......................................................................................... 61
# List of Figures

1. The TRIVAC modular approach. ......................................................... 2
2. Hexagonal geometries of type S30 and SA60 ....................................... 5
3. Hexagonal geometries of type SB60 and S90 ....................................... 6
4. Hexagonal geometries of type R120 and R180 ..................................... 6
5. Hexagonal geometry of type SA180 .................................................... 7
6. Hexagonal geometry of type SB180 .................................................... 7
7. Hexagonal geometry of type COMPLETE ........................................... 8
8. Cylindrical correction in Cartesian geometry ...................................... 8
9. Slab geometry with mesh-splitting .................................................... 12
10. Two-dimensional hexagonal geometry .............................................. 12
11. Description of the IAEA-2D benchmark. .......................................... 48
12. Description of the Biblis-2D benchmark, rods-withdrawn configuration. 50
13. Description of the IAEA-3D benchmark. ........................................... 52
14. Description of the S30 hexagonal benchmark. ................................... 55
15. Description of the LMW benchmark in 2-D. ...................................... 56
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Structure (TRIVAC)</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Structure (GEO:)</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Structure (geo_data1)</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Structure (geo_data2)</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Structure (descBC)</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Structure (descMC)</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>Structure (descPOS)</td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>Structure (MAC:)</td>
<td>13</td>
</tr>
<tr>
<td>9</td>
<td>Structure (mac_data)</td>
<td>13</td>
</tr>
<tr>
<td>10</td>
<td>Structure (macxs)</td>
<td>15</td>
</tr>
<tr>
<td>11</td>
<td>Structure (BIVACT:)</td>
<td>18</td>
</tr>
<tr>
<td>12</td>
<td>Structure (bivact_data)</td>
<td>18</td>
</tr>
<tr>
<td>13</td>
<td>Structure (TRIVAT:)</td>
<td>21</td>
</tr>
<tr>
<td>14</td>
<td>Structure (trivat_data)</td>
<td>21</td>
</tr>
<tr>
<td>15</td>
<td>Structure (BIVACA:)</td>
<td>24</td>
</tr>
<tr>
<td>16</td>
<td>Structure (bivaca_data)</td>
<td>24</td>
</tr>
<tr>
<td>17</td>
<td>Structure (TRIVAA:)</td>
<td>25</td>
</tr>
<tr>
<td>18</td>
<td>Structure (trivaa_data)</td>
<td>25</td>
</tr>
<tr>
<td>19</td>
<td>Structure (FLUD:)</td>
<td>27</td>
</tr>
<tr>
<td>20</td>
<td>Structure (flud_data)</td>
<td>27</td>
</tr>
<tr>
<td>21</td>
<td>Structure (DELTA:)</td>
<td>30</td>
</tr>
<tr>
<td>22</td>
<td>Structure (delta_data)</td>
<td>30</td>
</tr>
<tr>
<td>23</td>
<td>Structure (GPTFLU:)</td>
<td>32</td>
</tr>
<tr>
<td>24</td>
<td>Structure (gptflu_data)</td>
<td>32</td>
</tr>
<tr>
<td>25</td>
<td>Structure (OUT:)</td>
<td>35</td>
</tr>
<tr>
<td>26</td>
<td>Structure (out_data)</td>
<td>35</td>
</tr>
<tr>
<td>27</td>
<td>Structure (ERROR:)</td>
<td>37</td>
</tr>
<tr>
<td>28</td>
<td>Structure (INIKIN:)</td>
<td>39</td>
</tr>
<tr>
<td>29</td>
<td>Structure (inikin_data)</td>
<td>39</td>
</tr>
<tr>
<td>30</td>
<td>Structure (KINSOL:)</td>
<td>43</td>
</tr>
<tr>
<td>31</td>
<td>Structure (kinsol_data)</td>
<td>44</td>
</tr>
<tr>
<td>32</td>
<td>Structure (VAL:)</td>
<td>46</td>
</tr>
<tr>
<td>33</td>
<td>Structure (descval)</td>
<td>46</td>
</tr>
</tbody>
</table>
1 INPUT DATA SPECIFICATIONS

1.1 Syntactic rules for input data specifications

The input data to any module is read in free format using the subroutine REDGET. The rules for specifying the input data are therefore given in this section. The users guide was written using the following conventions:

- the parameters surrounded by single square brackets ‘[ ]’ denote an optional input;
- the parameters surrounded by double square brackets ‘[[ ]]’ denote an optional input which may be repeated as many times as desired;
- the parameters in braces separated by vertical bars ‘{ | | }’ denote a choice of input where (one and only one is mandatory);
- the parameters in **bold face** and in brackets ‘( )’ denote an input structure;
- the parameters in italics and in brackets with an index ‘(data(i), i=1,n)’ denote a set of n inputs;
- the words using the typewriter font are character constants **keywords** used as keywords;
- the words in italics are user defined variables, they should be lower case and are of type integer (starting with i to n) and real (starting with a to h or o to z) or of type character in uppercase **CHARACTER**.

1.2 The global input structure

TRIVAC is built around the GAN generalized driver.[3] Input data must therefore follow the calling specifications given below:

Table 1: Structure (TRIVAC)

```
[ LINKED_LIST [[ NAME1 ]] ; ]
[ XSM_FILE [[ NAME2 ]] ; ]
[ SEQ_BINARY [[ NAME3 ]] ; ]
[ SEQ_ASCII [[ NAME4 ]] ; ]
[ MODULE [[ NAME5 ]] ; ]
[(specif) ]
END: ;
```

where

- **NAME1** Character*12 name of a LCM object.
- **NAME2** Character*12 name of an XSM file.
- **NAME3** Character*12 name of a sequential binary file.
- **NAME4** Character*12 name of a sequential ASCII file.
- **NAME5** Character*12 name of a module.
- **(specif)** Input specifications for a single module. Specifications for TRIVAC modules will be given in the following sections.
The input data always begin with the declaration of each LCM object, XSM file, sequential (binary or ASCII) file that will be required by the following modules. This is followed by the declaration of the modules actually used in the input data deck. The following data describe a sequence of module calls, in the format of the GAN generalized driver. As indicated in Fig. 1, the modules communicate with each other through LCM objects or XSM files whose specifications are given in section 2. The TRIVAC user generally have the choice to declare its data structures as LINKED LIST to reduce CPU time resources or as XSM_FILE to reduce CPU memory resources.

The input data always end with a call to the END: module.

![Figure 1: The TRIVAC modular approach.](image)

### 1.3 The GEO: module

The GEO: module is used to create or modify a geometry. The geometry definition module in TRIVAC permits all the characteristics (coordinates, material mixture type indices and boundary conditions) of a simple or complex geometry to be specified. The method used to specify the geometry is independent of the discretization module to be used subsequently. Each geometry is represented by a name (character*12) and is saved in a LCM object or an XSM file under its given name. It is always possible to modify a given existing geometry or copy it into a neighbouring LCM object under a new name. The calling specifications are:

**Table 2: Structure (GEO:)**

```plaintext
{ GEO1 := GEO: :: (geo_data1) | GEO1 := GEO: { GEO1 | GEO2 } :: (geo_data2) }
```

where

- **GEO1** character*12 name of the LCM object (type L_GEO) that will contain the geometry.
- **GEO2** character*12 name of a LCM object (type L_GEO) containing the existing geometry. The type and all the characteristics of GEO2 will be copied onto GEO1.
(geo_data1) structure describing the characteristics of a new geometry (see Sect. 1.3.1).
(geo_data2) structure describing the change to the characteristics of an existing geometry (see Sect. 1.3.1).

1.3.1 Data input for module GEO:

Structures (geo_data1) and (geo_data2) serve to define the principle components of a geometry (dimensions, materials, boundary conditions):

Table 3: Structure (geo_data1)

<table>
<thead>
<tr>
<th>HOMOGE</th>
<th>CAR1D lx</th>
<th>TUBE lr</th>
<th>SPHERE lr</th>
<th>CAR2D lx ly</th>
<th>TUBEZ lr lz</th>
<th>CAR3D lx ly lz</th>
<th>HEX lh</th>
<th>HEXZ lh lz</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDIT</td>
<td>iprint</td>
<td>(descBC)</td>
<td>(descMC)</td>
<td>(descPOS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Structure (geo_data2)

| EDIT | iprint |
|      |        |
| (descBC) | (descMC) | (descPOS) |
| ;      |         |

where

HOMOGE    infinite homogeneous geometry.
CAR1D     one dimensional plane geometry (infinite slabs).
TUBE      cylindrical geometry (infinite tubes or cylinders).
SPHERE    spherical geometry (concentric spheres).
CAR2D     two-dimensional cartesian geometry.
TUBEZ     polar geometry ($R - Z$).
CAR3D     three-dimensional cartesian geometry.
HEX       two-dimensional hexagonal geometry.
HEXZ      three-dimensional hexagonal geometry.
lx        number of subdivisions along the X axis (before mesh-splitting).
ly number of subdivisions along the Y axis (before mesh-splitting).

lz number of subdivisions along the Z axis (before mesh-splitting).

lr number of cylinders or spherical shells (before mesh-splitting).

lh number of hexagons in an axial plane (including the virtual hexagons).

EDIT keyword used to set iprint.

iprint index used to control the printing in module GEO: =0 for no print; =1 for minimum printing (default value); =2 for printing the geometry state vector.

(descBC) structure allowing the boundary conditions surrounding the geometry to be treated.

(descMC) structure allowing material mixtures to be associated with a geometry.

(descPOS) structure allowing the coordinates of a geometry to be described.

The inputs corresponding to the (descBC) structure are the following:

Table 5: Structure (descBC)

| [ X- { VOID | REFL | DIAG | TRAN | SYME | ALBE { albedo | icode } | ZERO ] |
| [ CYLI | ACYL { albedo | icode } ] ] |
| [ X+ { VOID | REFL | DIAG | TRAN | SYME | ALBE { albedo | icode } | ZERO ] |
| [ CYLI | ACYL { albedo | icode } ] ] |
| [ Y- { VOID | REFL | DIAG | TRAN | SYME | ALBE { albedo | icode } | ZERO ] |
| [ CYLI | ACYL { albedo | icode } ] ] |
| [ Y+ { VOID | REFL | DIAG | TRAN | SYME | ALBE { albedo | icode } | ZERO ] |
| [ CYLI | ACYL { albedo | icode } ] ] |
| [ Z- { VOID | REFL | TRAN | SYME | ALBE { albedo | icode } | ZERO } ] |
| [ Z+ { VOID | REFL | TRAN | SYME | ALBE { albedo | icode } | ZERO } ] |
| [ R+ { VOID | REFL | ALBE { albedo | icode } | ZERO } ] ] |
| [ HBC [ S30 | SA60 | SB60 | S90 | R120 | R180 | SA180 | SB180 | COMPLETE ] ] |
| [ VOID | REFL | SYME | ALBE { albedo | icode } | ZERO } ] |
| [ RADS [ ANG ] nrads (xrad(ir), rrad(ir), [, ang(ir) ], ir=1,nrads ) ] ] |

where

X- negative X side.

Y- negative Y side.

Z- negative Z side.

X+ positive X side.

Y+ positive Y side.

Z+ positive Z side.

R+ side surrounding cylinders or spheres.

HBC side surrounding a hexagonal geometry.

VOID the side under consideration has a zero incoming current boundary condition.
the side under consideration has a reflective boundary condition.

the side under consideration is external to a diagonal axis of symmetry.

the side under consideration is connected to the opposite side of the domain. This option permits a translation condition to be treated.

the side under consideration is next to an axial axis of symmetry. (symmetric with respect to the central axis of the last row of volumes). The SYME condition can also be used in hexagonal geometry, but only with S30 and SA60 symmetries.

the side under consideration has an arbitrary albedo to be specified.

general geometrical albedo corresponding to the boundary condition ALBE \((\text{albedo} \geq 0.0)\).

index of a physical albedo corresponding to the boundary condition ALBE. The numerical values of the physical albedo are supplied by the module MAC:

the side under consideration has a zero flux boundary condition.

the side under consideration has a zero incoming current boundary condition with a circular correction applied on the Cartesian boundary. This option is only available in the \(X-Y\) plane for CAR2D and CAR3D geometries defined for TRIVAC full-core calculations.

the side under consideration has an arbitrary albedo with a circular correction applied on the Cartesian boundary. This option is only available in the \(X-Y\) plane for CAR2D and CAR3D geometries defined for TRIVAC full-core calculations.

hexagonal symmetry of one twelfth of an assembly (see Fig. 2).

hexagonal symmetry of one sixth of an assembly of type A (see Fig. 2).

hexagonal symmetry of one sixth of an assembly of type B (see Fig. 2).

hexagonal symmetry of one quarter of an assembly (see Fig. 2).

hexagonal symmetry of one third of an assembly (rotational symmetry) (see Fig. 2).

rotational symmetry of a half assembly (see Fig. 2).

hexagonal symmetry of half a type A assembly (see Fig. 2).

hexagonal symmetry of half a type B assembly (see Fig. 2).

complete hexagonal assembly (see Fig. 7).
Figure 3: Hexagonal geometries of type SB60 and S90

Figure 4: Hexagonal geometries of type R120 and R180
Figure 5: Hexagonal geometry of type SA180

Figure 6: Hexagonal geometry of type SB180
**RADS**

This keyword is used to specify the cylindrical correction applied in the *X* − *Y* plane for CAR2D and CAR3D geometries.\cite{footnote}

**ANG**

This keyword allows the angle (see Fig. 8) of the cylindrical notch to be set. By default, no notch is present.

*nrads*  
Number of different corrections along the cylinder main axis (i.e. the *Z* axis).

*xrad(ir)*  
Coordinate of the *Z* axis from which the correction is applied.

*rrad(ir)*  
Radius of the real cylindrical boundary.

*ang(ir)*  
Angle of the cylindrical notch. This data is given if and only if the keyword **ANG** is present.  
\( \text{ang(ir)} = \frac{\pi}{2} \) by default (i.e. the correction is applied at every angle).

---

The only combinations of diagonal symmetry permitted are: *X*+ **DIAG** *Y*− **DIAG** and *X*− **DIAG** *Y*+ **DIAG**. In these cases the geometry must be a square. The only combinations of translational symmetry permitted are: *X*− **TRAN** *X*+ **TRAN**, *Y*− **TRAN** *Y*+ **TRAN** and *Z*− **TRAN** *Z*+ **TRAN**.
The input corresponding to the (descMC) structure are the following:

Table 6: Structure (descMC)

```
[ MIX { (imix(i),i=1,lreg) | 
[[ PLANE iplan { (imix(i),i=1,lp) | SAME iplan1 
| [[ CROWN { (imix(i),i=1,lc) | ALL jmix | SAME iplan1 } ]] 
| [[ UPTO ic ALL jmix | SAME iplan1 } ]] } ] }
```

where

MIX
키워드로 각 볼륨의 재료 혼합 수를 각축의 대칭성 내에 각 볼륨에 할당하는 역할을 합니다. 볼륨이 대칭성의 내부에 있지만 계산 영역 밖에 위치할 경우 'افتراض적인' 볼륨으로 선언되어야 합니다 (예: 핵기구의 코너). 재료 혼합 수는 각 볼륨이 메시지분할이 이루어질 때만 지정되어야 합니다.

imix
재료 혼합의 유형과 각각의 영역에 대해서는 무시할 수 있습니다. imix ≤ nmixt 의 조건이 만족되며 nmixt는 모듈에서 정의됩니다. imix = 0의 경우 해당 볼륨은 VOID 경계 조건으로 대체됩니다. 이 경우 볼륨은 'افتراض적인' 볼륨으로 간주되어 환산은 계산되지 않습니다. 대각선 대칭계의 경우, 유형 지정은 대칭성의 외부를 포함하는 볼륨에는 지정할 수 없습니다. 이 값들은 선정시 주어진 유형이 되는 영역의 안쪽에서부터 외부까지의 순서대로 지정되어야 합니다: X-에서 X+, Y-에서 Y+, Z-에서 Z+ 그리고 대칭성의 중심에서부터 외곽에까지의 순서로.

PLANE
키워드로 각 2D 평면 내의 볼륨에 할당하는 재료 혼합 수를 할당하는 역할을 합니다. 이 옵션은 3D 모델의 평면에서만 사용할 수 있습니다. 이 옵션은 직선형 또는 육각형 모델에만 사용할 수 있습니다.

iplan
평면 번호에 대해서는 재료 혼합 수를 할당하는 볼륨에 대한 입력 메타데이터입니다.

SAME
키워드로 같은 재료 혼합 수를 iplan1 평면과 iplan 평면의 같은 부분에 할당하는 역할을 합니다. 이 키워드는 육각형 모델에서만 유효합니다. 현재 평면에서의 시계 방향의 조합의 재료 혼합에 대해서는 모든 재료 혼합 수의 동일성을 할당하는 역할을 합니다. 이 경우, 현재 평면에서의 모든 재료 혼합 수는 iplan1 평면에서의 동일한 재료 혼합 수와 일치할 것입니다.

iplan1
리포트 번호에 대해서는 현재 평면 또는 크라운(s)에 대한 입력 메타데이터를 할당하는 역할을 합니다.

lp
평면의 볼륨 수입니다. 직선형 모델에서 lp = lx * ly, 육각형 모델에서 lp = lh.

CROWN
키워드로 각 육각형의 한 육각형에 할당하는 재료 혼합 수를 할당하는 역할을 합니다. 이 옵션은 COMPLETE 육각형의 모델링에 대해서만 유효합니다. 이 키워드가 현재 크라운의 번호를 증가시키는 역할을 합니다. 또한, 이 키워드는 지정할 필요가 없습니다. 하지만 크라운을 지정할 때는 각 크라운의 중심 부근에서부터 외곽에까지의 순서대로 지정해야 합니다.

lc
현재 크라운의 육각형 수입니다. i번째 크라운의 완전한 육각형 평면의 경우, lc = (i − 1) * 6. 첫 번째 크라운은 만든 육각형의 수로 지정해야 합니다.

ALL
키워드로 lc 재료 혼합 수의 현재 크라운에 대해서는 동일한 재료 혼합 수 jmix를 지정하는 역할을 합니다.

UPTO
키워드로 현재 크라운까지의 재료 혼합 수를 할당하는 역할을 합니다. ic의 마지막 크라운에 대한 지정의 경우. 이 값은 현재 크라운의 번호보다 크거나 같아야 합니다.
Here we will assume that $l_{reg}$ is the exact number of cells or elementary cases to be considered. For example, if we had used the `DIAG` option with a geometry of type `CAR3D` ($lx=ly$), we would have: 

$$l_{reg}=(lx+1)*ly*lz/2.$$ 

The following dimensional constraints must also be respected:

- $n_{merge}$=number of merged cells (with $n_{merge} \geq l_{reg}$),
- $ngen$=number of generation cells (with $ngen \geq n_{merge}$),

The inputs corresponding to the `(descPOS)` structure are the following:

<table>
<thead>
<tr>
<th>Table 7: Structure (descPOS)</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>MESHX xxx(i),i=1,lx+1 ]</code></td>
</tr>
<tr>
<td><code>MESHY yyy(i),i=1,ly+1 ]</code></td>
</tr>
<tr>
<td><code>MESHZ zzz(i),i=1,lz+1 ]</code></td>
</tr>
<tr>
<td><code>RADIUS rrr(i),i=1,lr+1 ]</code></td>
</tr>
<tr>
<td><code>SIDE sidhex ]</code></td>
</tr>
<tr>
<td><code>SPLITX ispltx(i),i=1,lx ]</code></td>
</tr>
<tr>
<td><code>SPLITY isplty(i),i=1,ly ]</code></td>
</tr>
<tr>
<td><code>SPLITZ ispltz(i),i=1,lz ]</code></td>
</tr>
<tr>
<td><code>SPLITR ispltr(i),i=1,lr ]</code></td>
</tr>
<tr>
<td>`{ SPLITL ispltl</td>
</tr>
</tbody>
</table>

where

- `MESHX` keyword for the mesh of the geometry along the $X$ axis.
- `MESHY` keyword for the mesh of the geometry along the $Y$ axis.
- `MESHZ` keyword for the mesh of the geometry along the $Z$ axis.
- `RADIUS` keyword for the mesh of the geometry in the radial direction.
- `SIDE` keyword for the length of a side of a hexagon.
- `xxx` abscissa, corresponding to the limits of the regions making up the geometry. These values must be given in order, from $X_-$ to $X_+$. If the geometry presents a diagonal symmetry this data will also be used for the ordinate.
- `yyy` ordinate, corresponding to the limits of the regions making up the geometry. These values must be given in order, from $Y_-$ to $Y_+$.
- `zzz` height, corresponding to the limits of the regions making up the geometry. These values must be given in order, from $Z_-$ to $Z_+$.
- `rrr` Radii in the cases of cylindrical (`TUBE` or `TUBEZ`), spherical (`SPHERE`). It is important to note that we must have $rrr(1)=0.0$.
- `sidhex` length of a side of a hexagon.
- `SPLITX` keyword for mesh splitting of the geometry along the $X$ axis.
- `SPLITY` keyword for mesh splitting of the geometry along the $Y$ axis.
- `SPLITZ` keyword for mesh splitting of the geometry along the $Z$ axis.
**SPLITR** keyword for mesh splitting of the geometry in the radial direction.

**ispltx** number of sub-volumes that will be defined for each row of the volume along the X-axis. If the geometry presents a diagonal symmetry this input will also be used for the splitting along the Y-axis. By default, ispltx=1.

**isplty** number of sub-volumes that will be defined for each row of the volume along the Y-axis. If the geometry presents a diagonal symmetry this input will also be used for the splitting along the X-axis. By default, isplty=1.

**ispltz** number of sub-volumes that will be defined for each row of the volume along the Z-axis. By default, ispltz(i)=1.

**ispltr** the value of ispltr gives the number of sub-volumes that will be defined for each tube or each spherical shell. A negative value permits a splitting into equal sub-volumes; a positive value permits a splitting into equal sub-radius spacings. By default, ispltr=1.

**SPLITH** keyword to specify that a triangular mesh splitting of the hexagonal geometry is to be performed – for HEX and HEXZ type geometries.

**ispith** value of the triangular mesh splitting. The number of triangles per hexagon is given by $6 \times isplth^2$. ispith = 0 is used for full hexagon discretization.

**SPLITL** keyword to specify that a lozenge mesh splitting of the hexagonal geometry is to be performed – for HEX and HEXZ type geometries.

**ispltl** value of the lozenge splitting. The number of lozenges per hexagon is given by $3 \times ispltl^2$.

The user of the options described above should take care not to exceed the limits imposed by the amount of dynamically allocated memory available. For a pure geometry, let us define the variables lxp, lyp, lzp and lrp as:

$$
\begin{align*}
\text{lxp} &= \sum_{i=1}^{ir} \text{ispltx}(i) \\
\text{lyp} &= \sum_{i=1}^{iy} \text{isplty}(i) \\
\text{lzp} &= \sum_{i=1}^{iz} \text{ispltz}(i) \\
\text{lrp} &= \sum_{i=1}^{ir} \text{ispltr}(i)
\end{align*}
$$

thus, the limits that must be respected are the following:

- $\text{lxp} \geq \text{maxpts}$ for a CAR1D geometry.
- $\text{lh} \geq \text{maxpts}$ for a HEX geometry.
- $\text{lrp} \geq \text{maxpts}$ for the TUBE and SPHERE geometries.
- $\text{lxp} \times \text{lyp} \geq \text{maxpts}$ for the CAR2D geometry without diagonal symmetry.
- $\text{lxp} \times (\text{lyp} + 1)/2 \geq \text{maxpts}$ for the CAR2D geometry with diagonal symmetry.
- $\text{lrp} \times \text{lzp} \geq \text{maxpts}$ for the TUBEZ geometry.
- $\text{lxp} \times \text{lyp} \times \text{lzp} \geq \text{maxpts}$ for the CAR3D geometry without diagonal symmetry.
• \(lxp \times (lyp + 1) \times lzp/2 \geq \text{maxpts}\) for the CAR3D geometry with diagonal symmetry.
• \(lh \times lzp \geq \text{maxpts}\) for the HEXZ geometry.

1.3.2 Examples of geometries

We will now give a few examples which will permit users to better understand the procedure used to define the geometries in TRIVAC.

1. Slab geometry (see Fig. 9):

```
GEOMETRY1 := GEO: :: CAR1D 6
  X- VOID X+ ALBE 1.2
  MESHX 0.0 0.1 0.3 0.5 0.6 0.8 1.0
  SPLITX 2 2 2 1 2 1
  MIX 1 2 3 4 5 6

; β=0.0 β=1.2

```

Figure 9: Slab geometry with mesh-splitting

2. Two-dimensional hexagonal geometry (see Fig. 10):

```
GEOMETRY4 := GEO: :: HEX 12
  HBC S30 ALBE 1.6
  SIDE 1.3
  MIX 1 1 1 2 2 2 3 3 3 4 5 6

; β=1.6

```

Figure 10: Two-dimensional hexagonal geometry
1.4 The MAC: module

In TRIVAC the macroscopic cross sections and diffusion coefficients are read from the input data file using REDLEC. The general format of the data for the MAC: module in TRIVAC is the following:

Table 8: Structure (MAC:)

\[
\text{MACR1} := \text{MAC:} [ \{ \text{MACR1} | \text{MACR2} \} ] :: (\text{mac\_data})
\]

where

MACR1 character*12 name of the LCM object (type L\text{MACROLIB}) containing the new Macrolib produced by the module. A Macrolib contains macroscopic cross sections and diffusion coefficients. If MACR1 appears on both LHS and RHS, it is updated; otherwise, it is created. If MACR1 is created, all macroscopic cross sections and diffusion coefficients are first initialized to zero.

MACR2 character*12 name of the LCM object (type L\text{MACROLIB}) containing a read-only Macrolib. The information existing in MACR2 is copied into MACR1, but MACR2 is not modified.

(mac_data) structure containing the data to module MAC: (see Sect. 1.4.1).

1.4.1 Data input for module MAC:

Table 9: Structure (mac_data)

\[
\begin{align*}
\text{EDIT } & \text{iprint } \\
\text{NGRO } & \text{ngroup } \\
\text{NIFI } & \text{nifiss } \\
\text{DELP } & \text{ndel } \\
\text{ANIS } & \text{naniso } \\
\text{NMIX } & \text{nmixt } \\
\text{DELP } & \text{ndg } \\
\text{ANIS } & \text{naniso } \\
\text{ALBP } & \text{nalbp } ((albedp(ig,ia),ig=1,ngroup),ia=1,nalbp) \\
\text{READ INPUT } & \{ [(\text{macxs})] | \text{OLD (triv2)} | \text{DOLD (trip2)} \} \\
\{ \text{STEP } & \text{istep READ INPUT } [(\text{macxs})] \} \\
; & 
\end{align*}
\]

where

EDIT keyword used to set \text{iprint}.

iprint index used to control the printing in module MAC: =0 for no print. The macroscopic cross sections will be printed if the parameter \text{iprint} is greater than or equal to 2. The transfer cross sections will be printed if this parameter is greater than or equal to 3.
NGRO keyword used to define the number of energy groups. This data is given if and only if MACRI is created.

\textit{ngroup} the number of energy groups used for the calculations in TRIVAC.

NIFI keyword used to specify the maximum number of fissile spectrum associated with each mixture. Each fission spectrum generally represents a fissile isotope. This information is required only if MACLIB is created and the cross sections are taken directly from the input data stream.

\textit{nifiss} the maximum number of fissile isotopes per mixture. The default value is \textit{nifiss}=1.

DELP keyword used to specify the number of delayed neutron groups.

\textit{ndel} the number of delayed neutron groups. The default value is \textit{ndel}=0.

ANIS keyword used to specify the maximum level of anisotropy permitted in the scattering cross sections. This information is required only if MACLIB is created and the cross sections are taken directly from the input data stream.

\textit{naniso} number of Legendre orders for the representation of the scattering cross sections. The default value is \textit{naniso}=1 corresponding to the use of isotropic scattering cross sections.

NMIX keyword used to define the number of material mixtures. This data is given if and only if MACRI is created.

\textit{nmixt} the maximum number of material mixtures (a material mixture is characterized by a distinct set of macroscopic cross sections).

DELP keyword used to set \textit{ndg}. This data is used only if the fission spectrum $\chi_p$ is different from the delayed neutron spectrum $\chi_i$ for each precursor group $i$.

\textit{ndg} number of delayed neutron groups.

ANIS keyword used to specify the maximum level of anisotropy permitted in the diffusion cross sections. This data is given only if MACRI is created.

\textit{naniso} the maximum level of anisotropy. The default value is \textit{naniso}=1.

ALBP keyword used for the input of the physical albedos.

\textit{nalbp} the number of physical albedos per energy group.

\textit{albedp} multigroup physical albedo array (real numbers).

STEP keyword used to create a perturbation directory.

\textit{istep} the index of the perturbation directory.

READ keyword used to specify input of the cross section information from default input by REDLEC.

\textbf{(macxs)} structure describing the format used for reading the mixture cross sections and diffusion coefficients (or perturbation values of the cross sections and diffusion coefficients) from the input data file.

OLD keyword used to specify input of the cross section information from default input by REDLEC in the TRIVAC-2 format. The nuclear data will be translated into TRIVAC format and printed on the listing.

\textbf{(triv2)} structure describing the format used for reading the mixture cross sections and diffusion coefficients from the input data file in TRIVAC-2 format.
keyword used to specify perturbed input of the cross section information from default input by REDLEC in the TRIVAC-2 format. The perturbed nuclear data will be translated into TRIVAC format and printed on the listing.

structure describing the format used for reading the mixture values of the perturbed cross sections and diffusion coefficients from the input data file in TRIVAC-2 format.

1.4.2 Description of the nuclear data

Table 10: Structure (**macxs**)

<table>
<thead>
<tr>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIX matnum</td>
</tr>
<tr>
<td>{ NTOT0</td>
</tr>
<tr>
<td>[ NTOT1 (xssig1(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ TRANC (xsstra(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ NUIGSF ((xssigf(jf,jg), jg=1,ngroup), jf=1,nifiss)]</td>
</tr>
<tr>
<td>[ CHI ((xschi(jf,jg), jg=1,ngroup), jf=1,nifiss)]</td>
</tr>
<tr>
<td>[ FIXE (xsfixe(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ DIFF (diff(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ DIFFX (xdiffx(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ DIFFY (xdiffy(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ DIFFZ (xdiffz(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ NUIGRD (((xssigd(jf,idel,jg), jg=1,ngroup), idel=1,ndel), jf=1,nifiss)] ]</td>
</tr>
<tr>
<td>[ CHDL (((xschid(jf,idel,jg), jg=1,ngroup), idel=1,ndel), jf=1,nifiss)] ]</td>
</tr>
<tr>
<td>[ OVER ((overv(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ H-FACTOR (xhfact(jg), jg=1,ngroup) ]</td>
</tr>
<tr>
<td>[ SCAT ((nbscat(jl,jg), ilastg(jl,jg), (scat(jl,jg,ig), ig=1,nbscat(jl,jg)), jg=1,ngroup), jl=1,aniso ) ]</td>
</tr>
</tbody>
</table>

where

MIX keyword to specify that the macroscopic cross sections associated with a new mixture are to be read.

matnum identifier for the next mixture to be read. The maximum value permitted for this identifier is **nmixt**. When **matnum** is absent, the mixtures are numbered consecutively starting with 1 or with the last mixture number read either on the GOXS or the input stream.

NTOT0 keyword to specify that the total macroscopic cross sections for this mixture follow.

TOTAL alias keyword for **NTOT0**.

xssigt array representing the multigroup total macroscopic cross section (Σg in cm⁻¹) associated with this mixture.

NTOT1 keyword to specify that the P₁-weighted total macroscopic cross sections for this mixture follows.

xssig1 array representing the multigroup P₁-weighted total macroscopic cross section (Σ₁g in cm⁻¹) associated with this mixture.

TRANC keyword to specify that the transport correction macroscopic cross sections for this mixture follows.
xsstra array representing the multigroup transport correction macroscopic cross section ($\Sigma_{tc}^g$ in cm$^{-1}$) associated with this mixture.

NUSIGF keyword to specify that the macroscopic fission cross section multiplied by the average number of neutrons per fission for this mixture follows.

xssigf array representing the multigroup macroscopic fission cross section multiplied by the average number of neutrons per fission ($\nu\Sigma_f^g$ in cm$^{-1}$) for all the fissile isotopes associated with this mixture.

CHI keyword to specify that the fission spectrum for this mixture follows. By default, if CHI is not provided, all fission neutrons are emitted in group index 1 (fast group).

xschi array representing the multigroup fission spectrum ($\chi^g$) for all the fissile isotopes associated with this mixture.

FIXE keyword to specify that the fixed neutron source density for this mixture follows.

xsfixe array representing the multigroup fixed neutron source density for this mixture ($S^g$ in s$^{-1}$cm$^{-3}$).

DIFF keyword to specify that the isotropic diffusion coefficient for this mixture follows.

diff array representing the multigroup isotropic diffusion coefficient for this mixture ($D^g$ in cm).

DIFFX keyword for input of the $X$–directed diffusion coefficient.

xdiffx array representing the multigroup $X$–directed diffusion coefficient ($D^g_x$ in cm) for the mixture matnum.

DIFFY keyword for input of the $Y$–directed diffusion coefficient.

xdiffy array representing the multigroup $Y$–directed diffusion coefficient ($D^g_y$ in cm) for the mixture matnum.

DIFFZ keyword for input of the $Z$–directed diffusion coefficient.

xdiffz array representing the multigroup $Z$–directed diffusion coefficient ($D^g_z$ in cm) for the mixture matnum.

NUSIGD keyword to specify that the delayed macroscopic fission cross section multiplied by the average number of neutrons per fission for this mixture follows.

xssigid array representing the delayed multigroup macroscopic fission cross section multiplied by the average number of neutrons per fission ($\nu\Sigma_{f,idel}^g$ in cm$^{-1}$) for all the fissile isotopes associated with this mixture.

CHDL keyword to specify that the delayed fission spectrum for this mixture follows.

xschid array representing the delayed multigroup fission spectrum ($\chi_{g,idel}^g$) for all the fissile isotopes associated with this mixture.

OVERV keyword for input of the multigroup average of the inverse neutron velocity.

overv array representing the multigroup average of the inverse neutron velocity ($<1/v_{g,m}>$) for the mixture matnum.

H–FACTOR keyword to specify that the power factor for this mixture follows.

hfact array representing the multigroup power factor for this mixture ($H^g$ in MeV cm$^{-1}$).

SCAT keyword to specify that the macroscopic scattering cross section matrix for this mixture follows.
array representing the number of secondary groups $ig$ with non-vanishing macroscopic scattering cross section towards the primary group $jg$ considered for each anisotropy level associated with this mixture.

array representing the group index of the most thermal group with non-vanishing macroscopic scattering cross section towards the primary group $jg$ considered for each anisotropy level associated with this mixture.

array representing the multigroup macroscopic scattering cross section ($\Sigma_{sl}^{ig\rightarrow jg}$ in cm$^{-1}$) from the secondary group $ig$ towards the primary group $jg$ considered for each anisotropy level associated with this mixture. The elements are ordered using decreasing secondary group number $ig$, from $ilastg$ to ($ilastg - nbcat + 1$), and an increasing primary group number $jg$.

For example, the two group isotropic and linearly anisotropic scattering cross sections ($ngroup=2$, $naniso=2$) given by:

\[
\begin{align*}
    & L & \Sigma_{s,l}^{1\rightarrow 1} & \Sigma_{s,l}^{1\rightarrow 2} & \Sigma_{s,l}^{2\rightarrow 1} & \Sigma_{s,l}^{2\rightarrow 2} \\
    & 0 & 0.50 \text{ cm}^{-1} & 0.20 \text{ cm}^{-1} & 0.03 \text{ cm}^{-1} & 0.40 \text{ cm}^{-1} \\
    & 1 & 0.05 \text{ cm}^{-1} & 0.00 \text{ cm}^{-1} & 0.00 \text{ cm}^{-1} & 0.04 \text{ cm}^{-1}
\end{align*}
\]

must be entered as:

\begin{verbatim}
SCAT (*L=0*) 2 2 (*2->1*) 0.03 (*1->1*) 0.50 \\
            2 2 (*2->2*) 0.40 (*1->2*) 0.20 \\
            (*L=1*) 1 1 (*1->1*) 0.05 \\
            1 2 (*2->2*) 0.04
\end{verbatim}
1.5 The BIVACT: module

The BIVACT: module is used to perform a BIVAC-type tracking on a 1D/2D geometry.[3,4,15] The geometry is analyzed and a LCM object with signature L.BIVAC is created with the following information:

- Diagonal and hexagonal symmetries are unfolded and the mesh-splitting operations are performed. Volumes, material mixture and averaged flux recovery indices are computed on the resulting geometry.
- A finite element discretization is performed and the corresponding numbering is saved.
- The unit finite element matrices (mass, stiffness, etc.) are recovered.

The calling specifications are:

Table 11: Structure (BIVACT::)

\[
\text{TRACK} := \text{BIVACT:} \ [ \text{TRACK} \ ] \ \text{GEOM} :: (\text{bivact}\_\text{data})
\]

where

TRACK character*12 name of the LCM object (type L.BIVAC) containing the tracking information. If TRACK appears on the RHS, the previous settings will be applied by default.

GEOM character*12 name of the LCM object (type L.GEOM) containing the geometry.

(bivact_data) structure containing the data to module BIVACT: (see Sect. 1.5.1).

1.5.1 Data input for module BIVACT:

Table 12: Structure (bivact_data)

\[
[ \ \text{EDIT} \ i\text{print} \ ] \\
[ \ \text{TITL} \ \text{TITLE} \ ] \\
[ \ \text{MAXR} \ \text{maxpts} \ ] \\
[ \ \{ \ \text{PRIM} \ [ \ \text{ielem icol} \ ] \\
\ | \ \text{DUAL} \ [ \ \text{ielem icol} \ ] \\
\ | \ \text{MCFD} \ \} \ ] \\
[ \ \{ \ \text{PN} \ | \ \text{SPN} \ \} \ n \ [ \ \text{SCAT} \ [ \ \text{DIFF} \ i\text{scat} \ ] \ [ \ \text{VOID} \ n\text{vd} \ ] \ ] \\
; 
\]

where

EDIT keyword used to set i\text{print}.

i\text{print} index used to control the printing in module BIVACT:: =0 for no print; =1 for minimum printing (default value); Larger values produce increasing amounts of output.
**TITL**

Keyword which allows the run title to be set.

**TITLE**

The title associated with a TRIVAC run. This title may contain up to 72 characters. The default when TITL is not specified is no title.

**MAXR**

Keyword which permits the maximum number of regions to be considered during a TRIVAC run to be specified.

**maxpts**

Maximum dimensions of the problem to be considered. The default value is set to the number of regions previously computed by the GEO module but this value is insufficient if symmetries or mesh-splitting are specified.

**PRIM**

Keyword to set a primal finite element (classical) discretization.

**DUAL**

Keyword to set a mixed-dual finite element discretization. If the geometry is hexagonal, a Thomas-Raviart-Schneider method is used.[10]

**MCFD**

Keyword to set a mesh-centered finite difference discretization in hexagonal geometry.

**ielem**

Order of the finite element representation. The values permitted are 1 (linear polynomials), 2 (parabolic polynomials), 3 (cubic polynomials) or 4 (quartic polynomials). By default $ielem=1$.

**icol**

Type of quadrature used to integrate the mass matrices. The values permitted are 1 (analytical integration), 2 (Gauss-Lobatto quadrature) or 3 (Gauss-Legendre quadrature). By default $icol=2$. The analytical integration corresponds to classical finite elements; the Gauss-Lobatto quadrature corresponds to a variational or nodal type collocation and the Gauss-Legendre quadrature corresponds to superconvergent finite elements.

**PN**

Keyword to set a spherical harmonics ($P_n$) expansion of the flux.[9]

**SPN**

Keyword to set a simplified spherical harmonics ($SP_n$) expansion of the flux.[9,10] This option is currently available with 1D and 2D Cartesian geometries and with 2D hexagonal geometries.

**n**

Order of the $P_n$ or $SP_n$ expansion (odd number). Set to zero for diffusion theory (default value).

**SCAT**

Keyword to limit the anisotropy of scattering sources.

**DIFF**

Keyword to force using $1/3D^9$ as $\Sigma_I^*$ cross sections. A $P_1$ or $SP_1$ method will therefore behave as diffusion theory.

**iscat**

Number of terms in the scattering sources. $iscat=1$ is used for isotropic scattering in the laboratory system. $iscat=2$ is used for linearly anisotropic scattering in the laboratory system. The default value is set to $n+1$ in $P_n$ or $SP_n$ case.

**VOID**

Keyword to set the number of base points in the Gauss-Legendre quadrature used to integrate void boundary conditions if $icol=3$ and $n \neq 0$.

**nvd**

Type of quadrature. The values permitted are: 0 (use a $(n+2)$-point quadrature consistent with $P_n$ theory), 1 (use a $(n+1)$-point quadrature consistent with $S_{n+1}$ theory), 2 (use an analytical integration of the void boundary conditions). By default $nvd=0$.

Various finite element approximations can be obtained by combining different values of $ielem$ and $icol$:

- **PRIM 1 1**: Linear finite elements;
- **PRIM 1 2**: Mesh corner finite differences;
- **PRIM 1 3**: Linear superconvergent finite elements;
- **PRIM 2 1**: Quadratic finite elements;
• PRIM 2 2 : Quadratic variational collocation method;
• PRIM 2 3 : Quadratic superconvergent finite elements;
• PRIM 3 1 : Cubic finite elements;
• PRIM 3 2 : Cubic variational collocation method;
• PRIM 3 3 : Cubic superconvergent finite elements;
• PRIM 4 2 : Quartic variational collocation method;
• DUAL 1 1 : Mixed-dual linear finite elements;
• DUAL 1 2 : Mesh centered finite differences;
• DUAL 1 3 : Mixed-dual linear superconvergent finite elements (numerically equivalent to PRIM 1 3);
• DUAL 2 1 : Mixed-dual quadratic finite elements;
• DUAL 2 2 : Quadratic nodal collocation method;
• DUAL 2 3 : Mixed-dual quadratic superconvergent finite elements (numerically equivalent to PRIM 2 3);
• DUAL 3 1 : Mixed-dual cubic finite elements;
• DUAL 3 2 : Cubic nodal collocation method;
• DUAL 3 3 : Mixed-dual cubic superconvergent finite elements (numerically equivalent to PRIM 3 3);
• DUAL 4 2 : Quartic nodal collocation method;
1.6 The TRIVAT: module

The TRIVAT: module is used to perform a TRIVAC-type tracking on a 1D/2D/3D geometry.\cite{4–8,15}

The geometry is analyzed and a LCM object with signature L\_TRIVAC is created with the following information:

- Diagonal and hexagonal symmetries are unfolded and the mesh-splitting operations are performed. Volumes, material mixture and averaged flux recovery indices are computed on the resulting geometry.
- A finite element discretization is performed and the corresponding numbering is saved.
- The unit finite element matrices (mass, stiffness, etc.) are recovered.
- Indices related to an ADI preconditioning with or without supervectorization are saved.

The calling specifications are:

```
Table 13: Structure (TRIVAT:)

TRACK := TRIVAT: [ TRACK ] GEOM :: (trivat_data)

```

where

- `TRACK` character\*12 of the LCM object (type L\_TRIVAC) containing the tracking information. If `TRACK` appears on the RHS, the previous settings will be applied by default.
- `GEOM` character\*12 of the LCM object (type L\_GEOM) containing the geometry.
- `(trivat_data)` structure containing the data to module TRIVAT: (see Sect. 1.6.1).

1.6.1 Data input for module TRIVAT:

```
Table 14: Structure (trivat_data)

[ EDIT iprint ]
[ TITL TITLE ]
[ MAXR maxpts ]
[ SPN n | SCAT [ DIFF | iscat ] | VOID nvd ]]
[ ADI nadi ]
[ VECT [ iseg | PRTV impv ] ]
;
```

where

- `EDIT` keyword used to set `iprint`. 
**iprint**

Index used to control the printing in module **TRIVAT**: 
- =0 for no print;
- =1 for minimum printing (default value);
- Larger values produce increasing amounts of output.

**TITL**

Keyword which allows the run title to be set.

**TITLE**

The title associated with a TRIVAC run. This title may contain up to 72 characters. The default when **TITL** is not specified is no title.

**MAXR**

Keyword which permits the maximum number of regions to be considered during a TRIVAC run to be specified.

**maxpts**

Maximum dimensions of the problem to be considered. The default value is set to the number of regions previously computed by the **GEO** module but this value is insufficient if symmetries or mesh-splitting are specified.

**PRIM**

Keyword to set a discretization based on the variational collocation method.

**DUAL**

Keyword to set a mixed-dual finite element discretization. If the geometry is hexagonal, a Thomas-Raviart-Schneider method is used.\(^{[16]}\)

**MCFD**

Keyword to set a discretization based on the nodal collocation method. The mesh centered finite difference approximation is the default option and is generally set using **MCFD**. The **MCFD** approximations are numerically equivalent to the **DUAL** approximations with \(icol=2\); however, the **MCFD** approximations are less expensive.

**LUMP**

Keyword to set a discretization based on the nodal collocation method with serendipity approximation. The serendipity approximation is different from the **MCFD** option in cases with \(ielem\geq2\). This option is not available for hexagonal geometries.

**ielem**

Order of the finite element representation. The values permitted are: 1 (linear polynomials), 2 (parabolic polynomials), 3 (cubic polynomials) or 4 (quartic polynomials). By default \(ielem=1\).

**icol**

Type of quadrature used to integrate the mass matrices. The values permitted are: 1 (analytical integration), 2 (Gauss-Lobatto quadrature) or 3 (Gauss-Legendre quadrature). By default \(icol=2\). The analytical integration corresponds to classical finite elements; the Gauss-Lobatto quadrature corresponds to a variational or nodal type collocation and the Gauss-Legendre quadrature corresponds to superconvergent finite elements.

**SPN**

Keyword to set a simplified spherical harmonics \((SP_n)\) expansion of the flux\(^{[9,10]}\). This option is available with 1D, 2D and 3D Cartesian geometries and with 2D and 3D hexagonal geometries.

**n**

Order of the \(P_n\) or \(SP_n\) expansion (odd number). Set to zero for diffusion theory (default value).

**SCAT**

Keyword to limit the anisotropy of scattering sources.

**DIFF**

Keyword to force using \(1/3D^g\) as \(\Sigma^g_1\) cross sections. A \(P_1\) or \(SP_1\) method will therefore behave as diffusion theory.

**iscat**

Number of terms in the scattering sources. \(iscat=1\) is used for isotropic scattering in the laboratory system. \(iscat=2\) is used for linearly anisotropic scattering in the laboratory system. The default value is set to \(n+1\) in \(P_n\) or \(SP_n\) case.

**VOID**

Keyword to set the number of base points in the Gauss-Legendre quadrature used to integrate void boundary conditions if \(icol=3\) and \(n \neq 0\).

**nvd**

Type of quadrature. The values permitted are: 0 (use a \((n+2)\)–point quadrature consistent with \(P_n\) theory), 1 (use a \((n+1)\)–point quadrature consistent with \(S_{n+1}\) theory), 2 (use an analytical integration of the void boundary conditions). By default \(nvd=0\).
ADI keyword to set the number of ADI iterations at the inner iterative level.

\textit{nadi} number of ADI iterations (default: \textit{nadi} = 2).

\textbf{VECT} keyword to set an ADI preconditionning with supervectorization. By default, TRIVAC uses an ADI preconditionning without supervectorization.

\textit{iseg} width of a vectorial register. \textit{iseg} is generally a multiple of 64. By default, \textit{iseg}=64.

\textbf{PRTV} keyword used to set \textit{impv}.

\textit{impv} index used to control the printing in supervectorization subroutines. =0 for no print; =1 for minimum printing (default value); Larger values produce increasing amounts of output.

Various finite element approximations can be obtained with different values of \textit{ielem} (see Sect. 1.5).
1.7 The BIVACA: module

The BIVACA: module is used to compute the finite element system matrices (type L\_SYSTEM) corresponding to a BIVAC tracking (type L\_BIVAC) and to a set of nuclear properties (type L\_MACROLIB). The calling specifications are:

Table 15: Structure (BIVACA:)

\[
\text{SYST} := \text{BIVACA: [ SYST ] MACRO TRACK :: (bivaca\_data)}
\]

where

- SYST character*12 name of the LCM object (type L\_SYSTEM) containing the system matrices. If SYST appears on the RHS, the system matrices previously stored in SYST are kept.
- MACRO character*12 name of the LCM object (type L\_MACROLIB) containing the macroscopic cross sections and diffusion coefficients.
- TRACK character*12 name of the LCM object (type L\_BIVAC) containing the BIVAC tracking.
- (bivaca\_data) structure containing the data to module BIVACA: (see Sect. 1.7.1).

1.7.1 Data input for module BIVACA:

Table 16: Structure (bivaca\_data)

\[
[ \text{EDIT} \ iprint] \ [ \text{UNIT} ]
\]

where

- EDIT keyword used to set iprint.
- iprint index used to control the printing in module BIVACA:. =0 for no print; =1 for minimum printing (default value); Larger values produce increasing amounts of output.
- UNIT A system matrix corresponding to cross sections all set to 1.0 is computed. This keyword is mandatory if the system matrices in SYST are going to be used by INIKIN: or KINSOL: modules (see Sects. 1.14 and 1.15).
1.8 The TRIVAA: module

The TRIVAA: module is used to compute the finite element system matrices (type L_SYSTEM) corresponding to a TRIVAC TRACKING (type L_TRIVAC) and to a set of nuclear properties (type L_MACROLIB). The calling specifications are:

Table 17: Structure (TRIVAA:)

\[
SYST := \text{TRIVAA:} \ [ \ SYST \ ] \ MACRO \ TRACK \ [ \ DMACRO \ ] :: \ (\text{trivaa\_data})
\]

where

- \textit{SYST} character*12 name of the LCM object (type L_SYSTEM) containing the system matrices. If \textit{SYST} appears on the RHS, the system matrices previously stored in \textit{SYST} are kept.
- \textit{MACRO} character*12 name of the LCM object (type L_MACROLIB) containing the macroscopic cross sections and diffusion coefficients.
- \textit{TRACK} character*12 name of the LCM object (type L_TRIVAC) containing the TRIVAC TRACKING.
- \textit{DMACRO} character*12 name of the LCM object (type L_MACROLIB) containing derivatives or perturbations of the macroscopic cross sections and diffusion coefficients. If \textit{DMACRO} is given, only the derivatives or perturbations of the system matrices are computed.

\textit{(trivaa\_data)} structure containing the data to module TRIVAA: (see Sect. 1.8.1).

1.8.1 Data input for module TRIVAA: 

Table 18: Structure (trivaa\_data)

\[
[ \ \text{EDIT} \ iprint \ ] \\
[ \ \text{SKIP} \ ] \ [ \ { \text{DERI} \ | \ \text{PERT} } \ ] \ [ \ \text{UNIT} \ ] \ [ \ \text{OVEL} \ ] \\
;
\]

where

- \textit{EDIT} keyword used to set \textit{iprint}.
- \textit{iprint} index used to control the printing in module TRIVAA:. =0 for no print; =1 for minimum printing (default value); Larger values produce increasing amounts of output.
- \textit{SKIP} keyword used to skip the system matrix assembly but to perform the \textit{L} – \textit{D} – \textit{L}^T factorization. Use the system matrices already present in \textit{SYST}.
- \textit{DERI} The information recovered from \textit{DMACRO} is used as derivatives of nuclear properties with respect to a state variable. Derivatives of system matrices with respect to the same state variable are computed.
The information recovered from _DMACRO_ is used as the perturbation of the nuclear properties. Perturbations of the system matrices are computed.

A system matrix corresponding to cross sections all set to 1.0 is computed. This keyword is mandatory if the system matrices in _SYST_ are going to be used by _INIKIN_: or _KINSOL_: modules (see Sects. 1.14 and 1.15).

The reciprocal neutron velocities for each material mixture are recovered from the input _MACROLIB_ _MACRO_ and used to compute the corresponding system matrices. This capability is deprecated.
1.9 The FLUD: module

The FLUD: module is used to compute the solution to an eigenvalue problem corresponding to a set of system matrices (type L.SYSTEM). The calling specifications are:

Table 19: Structure (FLUD:)

| FLUX := FLUD: [ FLUX | SYST TRACK | MACRO ] :: (flud_data) |

where

- FLUX character*12 name of the LCM object (type L.FLUX) containing the solution. If FLUX appears on the RHS, the solution previously stored in FLUX is used to initialize the new iterative process; otherwise, a uniform unknown vector is used.
- SYST character*12 name of the LCM object (type L.SYSTEM) containing the system matrices.
- TRACK character*12 name of the LCM object (type L.TRACK) containing the TRACKING.
- MACRO character*12 name of the optional LCM object (type L.MACROLIB) containing the cross sections. This object is only used to set a link to the MACROLIB name inside the FLUX object. By default, the name of the MACROLIB is recovered from the link in the SYSTEM object.
- (flud_data) structure containing the data to module FLUD: (see Sect. 1.9.1).

1.9.1 Data input for module FLUD:

Table 20: Structure (flud_data)

| [ EDIT iprint ] |
| [ { VAR1 | ACCE } icl1 icl2 ] [ IRAM blisz korg [ nstard [ EPSG epsmsr ] ] ] |
| [ EXTE [ maxout ] [ epsout ] ] |
| [ THER [ maxthr ] [ epsthr ] ] |
| [ ADI nadi ] |
| [ ADJ ] |
| [ MONI lmod [ RAND ] ] |
| [ RELAX relax ] |

where

- EDIT keyword used to set iprint.
- iprint index used to control the printing in module FLUD:. =0 for no print; =1 for minimum printing (default value); =2 iteration history is printed; =3 the solution is printed; =4 at each iteration, the new solution is compared to a reference solution previously stored in FLUX under name REF; =5 the convergence histogram is stored in FLUX.
VAR1: keyword used to set the parameters $icl1$ and $icl2$. These parameters are used with the symmetrical variational acceleration technique (SVAT) for convergence of the generalized eigenvalue problem (default option) and to accelerate up-scattering iterations.

ACCE: alias keyword for VAR1.

$icl1$: number of free outer iterations in a cycle of the variational acceleration technique. The default value is $icl1 = 3$.

$icl2$: number of accelerated outer iterations in a cycle of the variational acceleration technique. The default value is $icl2 = 3$. A convergence in free iterations is obtained by setting $icl1 = 200$ (or $icl1 = maxout$) and $icl2 = 0$.

IRAM: keyword used to switch on the implicit restarted Arnoldi method (IRAM) and to set the parameters $blsz$, $korg$ and $nstard$.[11] By default, the symmetrical variational acceleration technique (SVAT) is used.

$blsz$: block size of the Arnoldi Hessenberg matrix. $blsz$ is the number of fixed-source Boltzmann transport equations solved simultaneously at each iteration of the implicit restarted Arnoldi method. The recommended value is $blsz = 3$.

$korg$: number of desired eigenvalues with $korg \geq bsz$.

$nstard$: number of iterations before restarting with the GMRES(m) acceleration method for solving the ADI-preconditioned linear systems in Trivac. The maximum number of GMRES iterations is set to $nadi$. By default, GMRES(m) acceleration is not used and $nadi$ free iterations are performed.

EPSG: keyword to specify the inner iteration GMRES epsilon.

$epsmsr$: convergence criterion for the inner iteration GMRES iterations. The fixed default value is $epsmsr = 1.0 \times 10^{-6}$.

EXTE: keyword to specify that the control parameters for the external iteration are to be modified.

$maxout$: maximum number of external iterations. The fixed default value is $maxout = 200$.

$epsout$: convergence criterion for the external iterations. The fixed default value is $epsout = 1.0 \times 10^{-4}$. The outer iterations are stopped when the following criteria is reached:

$$\max_{i} |\Phi^{(k-1)}_{i} - \Phi^{(k)}_{i}| \leq epsout \times \max_{i} |\Phi^{(k)}_{i}|$$

where $\Phi^{(k)} = \text{col}\{\Phi^{(k)}_{i}; i = 1, I\}$ is the product of the $B$ matrix times the unknown vector at the $k$-th outer iteration.

THER: keyword to specify that the control parameters for the thermal iterations are to be modified.

$maxthr$: maximum number of thermal iterations. The fixed default value is $maxthr = 0$ corresponding to no thermal iterations.

$epsthr$: convergence criterion for the thermal iterations. The fixed default value is $epsthr = 1.0 \times 10^{-5}$.

ADI: keyword used to set the number of alternating direction implicit (ADI) inner iterations in cases where Trivac is used. This keyword is also used to set the number of flux iterations over Legendre orders with $SP_n$ Bivac and Trivac cases if $n \geq 3$.

$nadi$: number of ADI inner or Legendre order iterations per outer iteration. The default value is $nadi = 1$. If this value causes a failure of the acceleration process, it is recommended that a larger value be tried. The optimal choice is generally the minimum value of $nadi$ which allows a convergence in less than 75 outer iterations. $nadi = 1$ or $nadi = 2$ is generally
the best choice for production-type calculations. The greater \( nadi \) is, the smaller the asymptotic convergence constant (ACC) becomes. Taking an arbitrary large value (e.g., \( nadi = 20 \)) leads to numerical results identical to those of the inverse power method where the system matrices are accurately inverted at each outer iteration (at a prohibitive CPU cost). In this case, the ACC is almost equal to the dominance ratio of the iterative matrix. The default value is recovered in the state vector of the \textit{TRACKING} object \textit{TRACK}.

\textbf{ADJ} \hspace{1cm} \textit{keyword used to obtain the solution to both the direct and adjoint eigenvalue problems. The adjoint solution is required if we subsequently want to perform a perturbation calculation. This option is limited to Trivac.}

\textbf{MONI} \hspace{1cm} \textit{keyword used to obtain the first harmonics of the solution and to set lmod. A full core representation of the reactor should be used to compute its harmonics. If symmetries are set in the geometry, some harmonics may be skipped. If the reactor is symmetric, a uniform initial estimate of the harmonics may cause some harmonics to be skipped; the keyword \textbf{RAND} should therefore be used.}

\textit{lmod} \hspace{1cm} \textit{the lmod first bi-orthonormalized harmonics of the solution are computed using the SVAT-accelerated preconditioned power method with a Hotelling deflation procedure.}\textsuperscript{12}

\textbf{RAND} \hspace{1cm} \textit{keyword used to initialize the harmonics calculations (option \textbf{MONI}) with a random estimate rather than a uniform estimate. This option has no effect if \textit{FLUX} appears on the RHS.}

\textbf{RELAX} \hspace{1cm} \textit{keyword used to set the relaxation parameter. This keyword must be specified each time a relaxation is required.}

\textit{relax} \hspace{1cm} \textit{relaxation parameter selected in the interval 0 < relax ≤ 1.0 and used to update the flux information in the \textit{FLUX} object. The updated value is taken equal to (1.0−relax) times the previous value (given in the RHS \textit{FLUX} object) plus relax times the value computed within current \textit{FLUD}: call. The default value is relax = 1.0.}
1.10 The $\textsc{Delta}$: module

The $\textsc{Delta}$: module is used to compute the source components of a fixed source eigenvalue problem corresponding to a set of unperturbed and perturbation system matrices (type $\text{L}_\text{System}$).

In the direct case, the fixed source is computed as:

$$\mathbf{S} = (\delta \mathbf{A} - \lambda_o \delta \mathbf{B}) \mathbf{\Phi} - \delta \lambda \mathbf{B}_o \mathbf{\Phi}$$

where the direct source vector $\mathbf{S}$ is orthogonal to the unperturbed adjoint flux $\mathbf{\Phi}^*$.

In the adjoint case, the fixed source is computed as:

$$\mathbf{S}^* = (\delta \mathbf{A}^\top - \lambda_o \delta \mathbf{B}^\top) \mathbf{\Phi}^* - \delta \lambda \mathbf{B}_o^\top \mathbf{\Phi}^*$$

where the adjoint source vector $\mathbf{S}^*$ is orthogonal to the unperturbed direct flux $\mathbf{\Phi}$ and where $\delta \lambda$ is the perturbation of the eigenvalue, as computed from the Rayleigh ratio.

The calling specifications are:

Table 21: Structure (DELTA:)

<table>
<thead>
<tr>
<th>Structure (DELTA:)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$GPT := \text{DELTA:} [ \ GPT \ ] FLUX0 SYST0 DSYST TRACK :: (\text{delta_data})$</td>
</tr>
</tbody>
</table>

where

$GPT$ character$^{12}$ name of the LCM object (type $\text{L}_\text{Source}$) containing the fixed source. If $GPT$ appears on the RHS, this information is used to initialize the state vector.

$FLUX0$ character$^{12}$ name of the LCM object (type $\text{L}_\text{Flux}$) containing the unperturbed flux.

$SYST0$ character$^{12}$ name of the LCM object (type $\text{L}_\text{System}$) containing the unperturbed system matrices.

$DSYST$ character$^{12}$ name of the LCM object (type $\text{L}_\text{System}$) containing a perturbation to the system matrices.

$TRACK$ character$^{12}$ name of the LCM object (type $\text{L}_\text{Track}$) containing the TRACKING.

($\text{delta_data}$) structure containing the data to module $\text{DELTA:}$ (see Sect. 1.10.1).

1.10.1 Data input for module $\text{DELTA:}$

Table 22: Structure (delta_data)

```
[ EDIT iprint ]
[ ADJ ] ;
```

where
EDIT keyword used to set `iprint`.

`iprint` index used to control the printing in module `DELTAL`.

ADJ keyword used to set the source on an adjoint fixed source eigenvalue problem.
1.11 The GPTFLU: module

The GPTFLU: module is used to compute the solution to a fixed source eigenvalue problem corresponding to a set of unperturbed system matrices and sources vectors.

If $\vec{S}$ is the source term of the explicit generalized adjoint equation, this module will solve:

$$(A_o - \lambda_o B_o) \vec{\Gamma}_i = \vec{S}_i$$  \hspace{1cm} (1.3)

where the direct source vector $\vec{S}_i$ is orthogonal to the adjoint flux.

If $\vec{S}$ is the source term of the implicit generalized adjoint equation, this module will solve:

$$(A_o^\top - \lambda_o B_o^\top) \vec{\Gamma}_j^* = \vec{S}_j^*$$  \hspace{1cm} (1.4)

where the adjoint source vector $\vec{S}_j^*$ is orthogonal to the direct flux.

The calling specifications are:

Table 23: Structure (GPTFLU:)

| FLUX_GPT : = GPTFLU: [ FLUX_GPT ] GPT FLUXO SYST TRACK :: (gptflu_data) |

where

**FLUX_GPT** character*12 name of the LCM object (type L_FLUX) containing the GPT solution. If FLUX_GPT appears on the RHS, the solution previously stored in FLUX_GPT is used to initialize the new iterative process; otherwise, a uniform unknown vector is used.

**GPT** character*12 name of the LCM object (type L_SOURCE) containing the fixed sources.

**FLUX0** character*12 name of the LCM object (type L_FLUX) containing the unperturbed flux used to decontaminate the GPT solution.

**SYST** character*12 name of the LCM object (type L_SYSTEM) containing the unperturbed system matrices.

**TRACK** character*12 name of the LCM object (type L_TRACK) containing the TRACKING.

**gptflu_data** structure containing the data to module GPTFLU:

1.11.1 Data input for module GPTFLU:

Table 24: Structure (gptflu_data)

| [ EDIT iprint ] |
| [ { VAR1 | ACCE } icl1 icl2 ] [ GMRES nstart ] |
| [ EXTE maxout ] [ epsout ] |
| [ THER maxthr ] [ epsthhr ] |
| [ ADI nadi ] |
| [ { EXPLICIT | IMPLICIT } ] |
| FROM-TO { ALL | i_{src1} i_{src2} } |
where

**EDIT** keyword used to set *iprint*.

**iprint** index used to control the printing in module **GPTFLU**: =0 for no print; =1 for minimum printing (default value); =2 iteration history is printed; =3 the solution is printed; =4 at each iteration, the new solution is compared to a reference solution previously stored in FLUX_GPT under the name **REF**; =5 the convergence histogram is stored in FLUX_GPT.

**VAR1** keyword used to set the parameters *icl1* and *icl2*. These parameters are used with the variational acceleration technique for convergence of the fixed-source iterations (default option) and to accelerate up-scattering iterations.

**ACCE** alias keyword for **VAR1**.

**icl1** number of free outer iterations in a cycle of the variational acceleration technique. The default value is *icl1* = 3.

**icl2** number of accelerated outer iterations in a cycle of the variational acceleration technique. The default value is *icl2* = 3. A convergence in free iterations is obtained by setting *icl1* = 200 (or *icl1* = *maxout*) and *icl2* = 0.

**GMRES** keyword to switch on the GMRES(m) acceleration of the fixed-source iterations. By default, the variational acceleration technique is used.

**nstart** restarts the GMRES method every *nstart* outer iterations. The recommended value is *nstart* = 10.

**EXTE** keyword to specify that the control parameters for the external iteration are to be modified.

**maxout** maximum number of external iterations. The fixed default value is *maxout* = 200.

**epsout** convergence criterion for the external iterations. The fixed default value is *epsout* = 1.0×10^{-4}. The outer iterations are stopped when the following criteria is reached:

\[
\max_i |\Gamma_i^{(k-1)} - \Gamma_i^{(k)}| \leq \text{epsout} \times \max_i |\Gamma_i^{(k)}|
\]

where \(\tilde{\Gamma}^{(k)} = \text{col}\{\Gamma_i^{(k)}; i = 1, I}\) is the product of the \(B\) matrix times the unknown vector at the \(k\)-th outer iteration.

**THER** keyword to specify that the control parameters for the thermal iterations are to be modified.

**maxthr** maximum number of thermal iterations. The fixed default value is *maxthr* = 0 corresponding to no thermal iterations.

**epsthr** convergence criterion for the thermal iterations. The fixed default value is *epsthr* = 1.0×10^{-2}.

**ADI** keyword used to set *nadi* in cases where Trivac is used.

**nadi** number of alternating direction implicit (ADI) inner iterations per outer iteration. The default value is *nadi* = 1. If this value causes a failure of the acceleration process, it is recommended that a larger value be tried. The optimal choice is generally the minimum value of *nadi* which allows a convergence in less than 75 outer iterations. *nadi* = 1 or *nadi* = 2 is generally the best choice for production-type calculations. The greater *nadi* is, the smaller the asymptotic convergence constant (ACC) becomes. Taking an arbitrary large value (e.g., *nadi* = 20) leads to numerical results identical to those obtained by inverting the system matrices at each outer iteration (at a prohibitive CPU cost). In this case, the ACC is almost equal to the dominance ratio of the iterative matrix.

**EXPLICIT** keyword used to obtain the solution of an direct fixed source eigenvalue problem.
IMPLICIT keyword used to obtain the solution of an adjoint fixed source eigenvalue problem. If neither 'EXPLICIT' nor 'IMPLICIT' are provided the default value will be chosen as a function of $n_{\text{var}}$ and $n_{\text{cst}} + 1$.

FROM-TO keyword used to specify the numbers of the sources for which a generalized adjoint will be calculated.

ALL keyword used to recover all sources available in GPT.

$i_{\text{src}1}$ number of the first source.

$i_{\text{src}1}$ number of the last source.
1.12 The OUT: module

The OUT: module is used to compute the reaction rates and to store them in an extended MACROLIB (type L_MACROLIB) corresponding to a solution (type L_FLUX) of the matrix system. The calling specifications are:

Table 25: Structure (OUT:)

\[ MACRO2 := \text{OUT: FLUX TRACK MACRO GEOM :: (out\_data)} \]

where

- **MACRO2** character*12 name of the LCM object (type L_MACROLIB) containing the extended MACROLIB.
- **FLUX** character*12 name of the LCM object (type L_FLUX) containing a solution.
- **TRACK** character*12 name of the LCM object (type L_TRACK) containing a TRACKING.
- **MACRO** character*12 name of the LCM object (type L_MACROLIB) containing the reference MACROLIB.
- **GEOM** character*12 name of the LCM object (type L_GEOM) containing the reference GEOMETRY.
- **(out\_data)** structure containing the data to module OUT:.

1.12.1 Data input for module OUT:

Table 26: Structure (out\_data)

\[
[\text{EDIT \ ipt} \ ] \\
[\text{MODE \ im}\ ] \\
[\{ \text{DIRE \ | \ PROD} \}] \\
[\{ \text{POWR \ | \ SOUR \ snumb} \}] \\
[\text{COND} \ {\{ \text{NONE \ |} \ (icond(i), i=1,ngcond) \}}] \\
[\text{INTG} \ {\{ \text{NONE \ |} \ IN \ | \ MIX \ | \ (ihom(i), i=1,nreg) \}}] \\
; 
\]

where

- **EDIT** keyword used to set \ipt.
- **ipt** index used to control the printing in module OUT: \(=0\) for no print; \(=-1\) for minimum printing (default value).
- **MODE** keyword to specify the flux harmonic index \im.
- **im** index of the flux harmonic recovered by the OUT: module if the MONI keyword was set in module FLUD: (see Sect. 1.9.1). By default, it is assumed that the MONI keyword was not used.
DIRECT

use the direct flux to perform homogenization and/or condensation (default option).

PROD

use the product of adjoint and direct fluxes to perform homogenization and/or condensation.

POWR

keyword used to set power.

power

value of the power in MW used to normalize the flux. By default, the flux is not normalized.

SOUR

keyword used to set snumb.

snumb

number of source particles used to normalize the flux. By default, the flux is not normalized.

COND

keyword to specify that a group condensation of the flux is to be performed. By default, no group condensation of the flux is to be performed, so that $ngcond=ngroup$.

icond

array of increasing energy group limits that will be associated with each of the $ngcond$ condensed groups. We must have $ngcond \leq ngroup$. By default, if COND is set and icond is not set, all energy groups are condensed together.

NONE

keyword to specify that no group condensation of the flux is to be performed, so that $ngcond=ngroup$ (default option).

INTG

keyword used to compute the reaction rates.

NONE

keyword for computing the reaction rates on the geometry mesh (see Sect. 1.3.1) after mesh-splitting.

IN

keyword for computing the reaction rates on the geometry mesh (see Sect. 1.3.1) before mesh-splitting.

MIX

keyword for computing the reaction rates on the mixture mesh previously used to define the geometry (see Sect. 1.3.1) before mesh-splitting.

ihom

index of the homogenized region corresponding to the each region of the geometry (see Sect. 1.3.1) before mesh-splitting.
1.13 The ERROR: module

The ERROR: module is used to compare reaction rates contained into two extended MACROLIBS and to print statistics regarding the comparison.

The QUANDRY-type power densities are first compared. These power densities are defined by the following relation:

\[ P_{\text{quandry}}^i = \frac{\sum V_i P_i}{\sum V_i P_i} \]

where \( P_i \) is the total power and \( V_i \) is the volume of the region \( i \). The maximum and averaged errors are respectively defined by:

\[ \epsilon_{\text{max}} = \max_i \frac{|P_{\text{quandry}}^i - P_{\text{quandry}^*}^i|}{P_{\text{quandry}^*}^i} \]

and

\[ \bar{\epsilon} = \frac{1}{V_{\text{core}}} \sum_i \left[ \frac{|P_{\text{quandry}}^i - P_{\text{quandry}^*}^i|}{P_{\text{quandry}^*}^i} \right] V_i \]

where \( P_{\text{quandry}^*}^i \) is computed using the reference powers (stored in MACRO1) and \( V_{\text{core}} \) is the total volume of the regions where the power density is not equal to zero.

The normalized removal rates \( T_{i,g}^{\text{norm}} \) in each region \( i \) and energy group \( g \) are next computed using the following formula:

\[ T_{i,g} = (\Sigma_{i,g} - \Sigma_{\text{wi},g}) \phi_{i,g} V_i \]

\[ T_{i,g}^{\text{norm}} = \frac{1}{\sum_i \sum_g T_{i,g}} T_{i,g} \]

where \( \Sigma_{i,g} \) is the total macroscopic cross section, \( \Sigma_{\text{wi},g} \) is the within-group scattering cross section and \( \phi_{i,g} \) is the neutron flux. The maximum and averaged errors are respectively defined by:

\[ \epsilon_{\max}^g = \max_i \frac{|T_{i,g}^{\text{norm}} - T_{i,g}^{\text{norm}^*}|}{T_{i,g}^{\text{norm}^*}} \]

and

\[ \bar{\epsilon}^g = \frac{1}{N} \sum_i \left[ \frac{|T_{i,g}^{\text{norm}} - T_{i,g}^{\text{norm}^*}|}{T_{i,g}^{\text{norm}^*}} \right] \]

where \( T_{i,g}^{\text{norm}^*} \) is computed using the reference values (stored in MACRO1) and \( N \) is the total number of regions in the MACROLIB.

The calling specifications are:

Table 27: Structure (ERROR:)

```plaintext
ERROR: MACRO1 MACRO2 : : [ HREA hname ] [ NREG nreg ] ;
```

where
**MACRO1**  character*12 name of the LCM object (type L_MACROLIB) containing the extended MACROLIB used to compute the reference reaction rates.

**MACRO2**  character*12 name of the LCM object (type L_MACROLIB) containing the extended MACROLIB used to compute the approximate reaction rates.

**HREA**  keyword used to set the character name *hname*.

**hname**  character*8 name of the nuclear reaction used to compute the power map. By default, reaction H-FACTOR is used.

**NREG**  keyword used to set the *nreg* number.

**nreg**  integer number set to the number of regions used in statistics. By default, all available regions are used.
1.14 The INIKIN: module

The INIKIN: module is used to recover the steady-state solution and to initialize the kinetics parameters. The delayed neutron information can be provided directly from the input file or recovered from the MACROLIB data structure.

The initial precursor concentrations are obtained as a function of the steady-state solution. If \( \phi_g(r, t_0) \) is the initial flux in energy group \( g \) divided by \( k_{\text{eff}} \), the corresponding initial conditions of the precursors are obtained as

\[
c_{\ell}(r, t_0) = \frac{1}{\lambda_{\ell}} \sum_{h=1}^{G} \nu_{\Sigma_{it\ell,h}}(r) \phi_h(r, t_0); \quad \ell = 1, N_d.
\]

(1.5)

where \( \nu_{\Sigma_{it\ell,h}}(r) \) is \( \nu \) times the delayed macroscopic fission cross section in energy group \( h \) for precursor group \( \ell \).

The calling specifications are:

Table 28: Structure (INIKIN:)

| KINET := INIKIN: MACRO TRACK SYST FLUX :: (inikin_data) |

where

- **KINET** character*12 name of the LCM object (type L_KINET) to be created by the module.
- **MACRO** character*12 name of the LCM object (type L_MACROLIB) containing the MACROLIB information.
- **TRACK** character*12 name of the LCM object (type L_TRACK) containing the TRACKING information.
- **SYST** character*12 name of the LCM object (type L_SYSTEM) corresponding to MACROLIB MACRO and TRACKING TRACK.
- **FLUX** character*12 name of the LCM object (type L_FLUX) containing the initial steady-state solution.

**(inikin_data)** structure containing the data to module INIKIN: (see Sect. 1.14.1).

1.14.1 Data input for module INIKIN:

Table 29: Structure (inikin_data)

```plaintext
[ EDIT iprint ]
[ NGRP ngrp ]
NDEL ndg
[ BETA (beta(i), i=1,ndg) ]
[ LAMBDA (lambda(i), i=1,ndg) ]
[ CHID ((chid(i), i=1,ndg), j=1,ngrp] ]
```

continued on next page
The [NORM { fnorm | MAX | POWER-INI power }] structure is used to set the initial flux. The

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDIT</td>
<td>Keyword used to set the <code>iprint</code> index.</td>
</tr>
<tr>
<td><code>iprint</code></td>
<td>Integer index used to control the printing in module <code>INIKIN</code>. =0 for no print; =1 for minimum printing (default value); larger values of <code>iprint</code> will produce increasing amounts of output.</td>
</tr>
<tr>
<td>NGRP</td>
<td>Keyword used to set the <code>ngrp</code> number. By default, this information is recovered from the solution object <code>FLUX</code>.</td>
</tr>
<tr>
<td><code>ngrp</code></td>
<td>Integer total number of energy groups.</td>
</tr>
<tr>
<td>NDEL</td>
<td>Keyword used to set the <code>ndg</code> number.</td>
</tr>
<tr>
<td><code>ndg</code></td>
<td>Integer total number of the delayed neutron groups.</td>
</tr>
<tr>
<td>BETA</td>
<td>Keyword used to indicate the reading of <code>beta</code> values from the input file. If these values are not provided, they should be recorded in the MACROLIB data structure.</td>
</tr>
<tr>
<td><code>beta</code></td>
<td>Real array containing the delayed neutron fractions for each delayed group.</td>
</tr>
<tr>
<td>LAMBDA</td>
<td>Keyword used to indicate the reading of <code>lambda</code> values from the input file. If these values are not provided, they should be recorded in the MACROLIB data structure.</td>
</tr>
<tr>
<td><code>lambda</code></td>
<td>Real array containing the precursors decay constants for each delayed group.</td>
</tr>
<tr>
<td>CHID</td>
<td>Keyword used to indicate the reading of <code>chid</code> values from the input file. If these values are not provided, they should be recorded in the MACROLIB data structure.</td>
</tr>
<tr>
<td><code>chid</code></td>
<td>Real array representing the delayed multigroup fission spectrum.</td>
</tr>
<tr>
<td>NORM</td>
<td>Keyword used to normalize the initial flux. By default, the flux is not normalized.</td>
</tr>
<tr>
<td><code>fnorm</code></td>
<td>Real normalization factor.</td>
</tr>
<tr>
<td>MAX</td>
<td>Keyword used to set the flux normalization factor to $1/f_{\text{max}}$ where $f_{\text{max}}$ is the maximum flux in the core.</td>
</tr>
<tr>
<td>POWER-INI</td>
<td>Keyword used to set the flux normalization factor to a given value of the initial power.</td>
</tr>
<tr>
<td><code>power</code></td>
<td>Real initial power in MW.</td>
</tr>
</tbody>
</table>
1.15 The KINSOL: module

The KINSOL: module is used to solve the space-time neutron kinetics equations at current time step of transient.

1.15.1 The direct (forward) solution

We first consider the discretization of the legacy forward space-time kinetics equation. Several implicit numerical schemes are available for this purpose. Consider first the differential equation for precursor concentrations:

\[
\frac{\partial c_\ell(r, t)}{\partial t} + \lambda_\ell c_\ell(r, t) = \sum_{h=1}^{G} \nu \Sigma_{i\ell,h}^{\text{del}}(r) \phi_h(r, t); \quad \ell = 1, N_d.
\] (1.6)

Consider a solution between times \(t_{n-1}\) and \(t_n = t_{n-1} + \Delta t_n\). First, an analytic solution can be obtained by assuming a ramp variation of the fission reaction rates over time step \(\Delta t_n\). This solution is written

\[
c_\ell(r, t_n) = c_\ell(r, t_{n-1}) e^{-\lambda_\ell} \Delta t_n + \frac{F_\ell(r, t_{n-1})}{\lambda_\ell} \left[ \frac{1}{\lambda_\ell} \frac{1 - e^{-\lambda_\ell} \Delta t_n}{e^{-\lambda_\ell} \Delta t_n} \right]
\]

\[
+ \frac{F_\ell(r, t_n)}{\lambda_\ell} \left[ 1 - \frac{1}{\lambda_\ell} \frac{1 - e^{-\lambda_\ell} \Delta t_n}{e^{-\lambda_\ell} \Delta t_n} \right]
\] (1.7)

where the delayed fission reaction rates are defined as

\[
F_\ell(r, t_n) = \sum_{h=1}^{G} \nu \Sigma_{i\ell,h}^{\text{del}}(r) \phi_h(r, t_n) = \beta_\ell c_\ell(r, t_n).
\] (1.8)

An implicit theta solution is presented in Chapter 5 of Ref. 1. This solution is written

\[
c_\ell(r, t_n) = \left[ 1 - (1 - \Theta_p) \frac{\lambda_\ell \Delta t_n}{1 + \Theta_p \lambda_\ell \Delta t_n} \right] c_\ell(r, t_{n-1}) + \frac{F_\ell(r, t_{n-1})}{\lambda_\ell} \frac{(1 - \Theta_p) \lambda_\ell \Delta t_n}{1 + \Theta_p \lambda_\ell \Delta t_n}
\]

\[
+ \frac{F_\ell(r, t_n)}{\lambda_\ell} \left[ \Theta_p \frac{\lambda_\ell \Delta t_n}{1 + \Theta_p \lambda_\ell \Delta t_n} \right]
\] (1.9)

where \(\Theta_p\) is the theta-factor for precursors.

The fixed-source corresponding to the analytic solution for precursors is written

\[
S_g^{\text{exact}}(r, t_n) = \frac{1}{V_{n,g} \Delta t_n} \phi_g(r, t_{n-1}) + \sum_{\ell} \lambda_\ell \left[ 1 - \Theta_\ell + \Theta_\ell e^{-\lambda_\ell \Delta t_n} \right] \chi_{\ell,g}^{\text{del}}(r) c_\ell(r, t_{n-1})
\]

\[
+ (1 - \Theta_\ell) \left\{ \nabla \cdot D_g(r) \nabla \phi_g(r, t_{n-1}) - \Sigma_{eg}(r) \phi_g(r, t_{n-1}) \right\}
\]

\[
+ \sum_{h=1}^{G} \Sigma_{g\rightarrow h}(r) \phi_h(r, t_{n-1}) + \chi_g^{\text{ss}}(r) F(r, t_{n-1}) \}
\]

\[
- \sum_{\ell} \left[ 1 - \Theta_\ell - \Theta_\ell \left( \frac{1}{\lambda_\ell} \frac{1 - e^{-\lambda_\ell \Delta t_n}}{e^{-\lambda_\ell \Delta t_n}} \right) \right] \chi_{\ell,g}^{\text{del}}(r) F_\ell(r, t_{n-1})
\] (1.10)

where the steady-state fission reaction rates are defined as

\[
F(r, t_n) = \sum_{h=1}^{G} \nu \Sigma_{ih}(r) \phi_h(r, t_n).
\] (1.11)
The fixed-source corresponding to the implicit theta solution is presented in Chapter 5 of Ref. 1 and is written

\[ S_g^\Theta(r, t_n) = \frac{1}{V_{n,g} \Delta t_n} \phi_g(r, t_n) - \Theta_t \sum_{h=1}^G \sum_{h \neq g} \Sigma_{g-h}(r) \phi_h(r, t_n) + \Theta_t \Sigma_{rg}(r) \phi_g(r, t_n) \]

\[ = S_g^{\text{exact}}(r, t_n) + \Theta_t \sum_{h=1}^G \sum_{h \neq g} \Sigma_{g-h}(r) \phi_h(r, t_n) \]

\[ + \Theta_t \chi_g^{\text{ss}}(r) F(r, t_n) - \Theta_t \sum_{h=1}^G \chi_{g}^{\text{del}}(r) \frac{1}{1 + \Theta_p \lambda_t \Delta t_n} (1 - e^{-\lambda_t \Delta t_n}) F_h(r, t_n). \] (1.12)

The flux equation at end-of-step is now presented. The equation corresponding to the analytic solution for precursors is written

\[ \frac{1}{V_{n,g} \Delta t_n} \phi_g(r, t_n) - \Theta_t \nabla \cdot \mathbb{D}_g(r) \nabla \phi_g(r, t_n) + \Theta_t \Sigma_{rg}(r) \phi_g(r, t_n) \]

\[ = \sum_{h=1}^G \sum_{h \neq g} \Sigma_{g-h}(r) \phi_h(r, t_n) \]

\[ + \Theta_t \chi_g^{\text{ss}}(r) F(r, t_n) - \Theta_t \sum_{h=1}^G \chi_{g}^{\text{del}}(r) \frac{1}{1 + \Theta_p \lambda_t \Delta t_n} F_h(r, t_n). \] (1.13)

The equation corresponding to the implicit theta solution is presented in Chapter 5 of Ref. 1 and is written

\[ \frac{1}{V_{n,g} \Delta t_n} \phi_g(r, t_n) - \Theta_t \nabla \cdot \mathbb{D}_g(r) \nabla \phi_g(r, t_n) + \Theta_t \Sigma_{rg}(r) \phi_g(r, t_n) \]

\[ = \sum_{h=1}^G \sum_{h \neq g} \Sigma_{g-h}(r) \phi_h(r, t_n) \]

\[ + \Theta_t \chi_g^{\text{ss}}(r) F(r, t_n) - \Theta_t \sum_{h=1}^G \chi_{g}^{\text{del}}(r) \frac{1}{1 + \Theta_p \lambda_t \Delta t_n} F_h(r, t_n). \] (1.14)

1.15.2 The adjoint (backward) solution

The negative sign in front of the term \((1/v)\partial \phi^*/\partial t\) suggest some sort of backward approach to compute the importance (as opposed to the direct or forward approach for the direct neutron flux). Hence, while it is necessary to define an initial state of the system to solve the direct equations, solving the importance equations requires final conditions and to proceed backward with respect to time.

Discretization of the adjoint space-time kinetics equations using the implicit theta solution leads to the following equations. The solution for precursors is written

\[ c^*_\ell(r, t_{n-1}) = \frac{1}{1 - (1 - \Theta_p) \lambda_t \Delta t_n} \left[ \frac{1 - (1 - \Theta_p) \lambda_t \Delta t_n}{1 + \Theta_p \lambda_t \Delta t_n} \right] c^*_\ell(r, t_n) + \frac{(1 - \Theta_p) \lambda_t \Delta t_n}{1 + \Theta_p \lambda_t \Delta t_n} \sum_{h=1}^G \chi_{\ell,h}^{\text{del}}(r) \phi^*_h(r, t_n) \]

\[ + \frac{\Theta_p \lambda_t \Delta t_n}{1 + \Theta_p \lambda_t \Delta t_n} \sum_{h=1}^G \chi_{\ell,h}^{\text{del}}(r) \phi^*_h(r, t_{n-1}). \] (1.15)
The flux equation at beginning-of-step is written
\[
\frac{1}{V_{n,g} \Delta t_n} \phi_g^*(r, t_{n-1}) - \Theta_f \nabla \cdot \mathbb{D}_g(r) \nabla \phi_g^*(r, t_{n-1}) + \Theta_f \sum_{\ell} \phi_{\ell,g}(r, t_{n-1}) \\
= S_g^\Theta(r, t_{n-1}) + \Theta_f \sum_{h=1}^{G} \Sigma_{h-g}(r) \phi_h^*(r, t_{n-1}) \\
+ \Theta_f \sum_{h=1 \atop h \neq g}^{G} \nu \Sigma_{h}(r) \chi_h(r) - \sum_{\ell} \nu \Sigma_{\ell}^{\text{del}}(r) \chi_{\ell,h}^{\text{del}}(r) \frac{1}{1 + \Theta_p \lambda_\ell \Delta t_n} \phi_h^*(r, t_{n-1})
\]

where the fixed-source \( S_g^\Theta(r, t_{n-1}) \) is written
\[
S_g^\Theta(r, t_{n-1}) = \frac{1}{V_{n,g} \Delta t_n} \phi_g(r, t_n) + \sum_{\ell} \nu \Sigma_{\ell}^{\text{del}}(r) \left[ 1 - \Theta_f + \Theta_f \frac{1 - (1 - \Theta_p \lambda_\ell) \Delta t_n}{1 + \Theta_p \lambda_\ell \Delta t_n} \right] c_\ell^*(r, t_n)
\]

The equations corresponding to the analytic solution for precursors are obtained by replacing the following terms in Eqs. (1.15) to (1.17):
\[
\frac{1}{1 + \Theta_p \lambda_\ell \Delta t_n} \Rightarrow \frac{1 - \exp^{-\lambda_\ell \Delta t_n}}{\lambda_\ell \Delta t_n} \quad \text{and} \quad \frac{1 - (1 - \Theta_p \lambda_\ell \Delta t_n)}{1 + \Theta_p \lambda_\ell \Delta t_n} \Rightarrow \exp^{-\lambda_\ell \Delta t_n}.
\]

1.15.3 The calling specifications

The calling specifications are:

| KINET | KINSOL: KINET MACRO TRACK SYST [ MACRO]0 SYST]0 | : (kinsol_data) |

where

- **KINET** character*12 name of the LCM object (type L_KINET) in modification mode.
- **MACRO** character*12 name of the LCM object (type L_MACROLIB) containing the MACROLIB information corresponding to the current time step of a transient.
- **TRACK** character*12 name of the LCM object (type L_TRACK) containing the TRACKING information.
- **SYST** character*12 name of the LCM object (type L_SYSTEM) corresponding to MACROLIB MACRO and TRACKING TRACK.
MACRO_0 character*12 name of the LCM object (type L_MACROLIB) containing the MACROLIB information corresponding to the beginning-of-step conditions in case a ramp variation of the cross sections in set. Beginning-of-step conditions should not be confused with beginning-of-transient or initial conditions. By default, a step variation is set where cross sections are assumed constant and given by MACRO.

SYST_0 character*12 name of the LCM object (type L_SYSTEM) corresponding to MACROLIB MACRO_0 and TRACKING TRACK.

(kinsol_data) structure containing the data to module KINSOL: (see Sect. 1.15.4).

1.15.4 Data input for module KINSOL:

Table 31: Structure (kinsol_data)

```
[ EDIT iprint ]
DELTA delta
SCHEME FLUX [ TEXP ] { IMPLIC | CRANK | THETA ttflix }
PREC { IMPLIC | CRANK | EXPON | THETA ttprc }
{ [ VAR1 | ACCE } icl1 icl2 ]
{ [ EXTE [ maxout ] [ epsout ] ]
{ [ THER [ maxthr ] [ epsthr ] ]
[ ADI nadi ]
[ ADJ ]
[PICK >> power_out << ]
```

where

EDIT keyword used to set iprint index.

iprint integer index used to control the printing in module KINSOL: =0 for no print; =1 for minimum printing (default value); larger values of iprint will produce increasing amounts of output.

DELTA keyword used to set the delta value.

delta current time increment $\Delta t_n$ of transient.

SCHEME keyword used to indicate the temporal numerical schemes.

TEXP keyword used to enable the exponential transformation procedure on transient flux. Mixture- and group-dependent factors $\omega_{m,g}$ are set such that the flux at point $r$ is defined as

$$\phi_g(r, t) = e^{\omega_{m,g}t} \tilde{\phi}_g(r, t)$$

(1.19)

where $m$ is the mixture index corresponding to point $r$. Factors $\omega_{m,g}$ are initialized to zero by module INIKIN: and are recomputed at the end of each time step.

FLUX keyword used to select the temporal scheme for the fluxes equations.

PREC keyword used to select the temporal scheme for the precursors equations.
IMPLIC  keyword used to indicate the full implicit temporal scheme.
CRANK  keyword used to indicate the Crank-Nicholson temporal scheme.
EXPON  keyword used to indicate the analytical integration scheme for precursors equations.
THETA  keyword used to indicate the general temporal scheme according to the \( \theta \) method.
\ttflx value of \( \theta \) parameter \( \Theta_f \) for the flux equations. This value should be greater than 0.5 and less than 1.0.
\ttprc value of \( \theta \) parameter \( \Theta_p \) for the precursors equations. This value should be greater than 0.5 and less than 1.0.
VAR1  keyword used to switch on the variational acceleration technique and to set the parameters icl1 and icl2.
ACCE  alias keyword for VAR1.
icl1 number of free outer iterations in a cycle of the variational acceleration technique. The default value is icl1 = 3.
icl2 number of accelerated outer iterations in a cycle of the variational acceleration technique. The default value is icl2 = 3. A convergence in free iterations is obtained by setting icl1 = 200 (or icl1 = maxout) and icl2 = 0.
EXTE  keyword to specify that the control parameters for the external iteration are to be modified.
maxout maximum number of external iterations. The fixed default value is maxout = 200.
epsout convergence criterion for the external iterations. The fixed default value is epsout = 1.0 \( \times 10^{-4} \). The outer iterations are stopped when the following criteria is reached:
\[
\max_i |\Phi_i^{(k-1)} - \Phi_i^{(k)}| \leq \text{epsout} \times \max_i |\Phi_i^{(k)}|
\]
where \( \Phi_i^{(k)} = \text{col}\{\Phi_i^{(k)}; i = 1, I\} \) is the product of the \( B \) matrix times the unknown vector at the \( k \)-th outer iteration.
THER  keyword to specify that the control parameters for the thermal iterations are to be modified.
\maxthr maximum number of thermal iterations. The fixed default value is maxthr = 0 corresponding to no thermal iterations.
\epsththr convergence criterion for the thermal iterations. The fixed default value is epsththr = 1.0 \( \times 10^{-2} \).
ADI  keyword used to set nadi in cases where Trivac is used.
nadi number of alternating direction implicit (ADI) inner iterations per outer iteration. The default value is nadi = 1. If this value causes a failure of the acceleration process, it is recommended that a larger value be tried. The optimal choice is generally the minimum value of nadi which allows a convergence in less than 75 outer iterations. nadi = 1 or nadi = 2 is generally the best choice for production-type calculations. The greater nadi is, the smaller the asymptotic convergence constant (ACC) becomes. Taking an arbitrary large value (e.g., nadi = 20) leads to numerical results identical to those obtained by inverting the system matrices at each outer iteration (at a prohibitive CPU cost). In this case, the ACC is almost equal to the dominance ratio of the iterative matrix. The default value is recovered in the state vector of the TRACKING object TRACK.
ADJ  keyword used to perform an adjoint (backward) space-time kinetics calculation. By default, a direct (forward) space-time kinetics calculation is performed.
PICK  keyword used to recover the end-of-stage power (in MW) in a CLE-2000 variable.
\power_out character\*12 CLE-2000 variable name in which the extracted power value will be placed.
1.16 The VAL: module

The VAL: module supplies an interpolation of the flux in diffusion calculations for Cartesian geometries. The calling specifications are:

Table 32: Structure (VAL:)

\[
IFLU := \text{VAL: TRKNAM FLUNAM} :: (\text{descval})
\]

where

- **IFLU** character*12 name of the INTERPFLUX data structure (L_FVIEW signature) where the interpolated flux distribution will be stored.
- **TRKNAM** character*12 name of the read-only TRACKING data structure (L_TRACK signature) containing the tracking.
- **FLUNAM** character*12 name of the read-only FLUXUNK data structure (L_FLUX signature) containing a transport solution.
- **(descval)** structure containing the input data to this module to compute interpolated flux (see Section 1.16.1).

1.16.1 Data input for module VAL:

Table 33: Structure (descval)

\[
[\text{EDIT } iprint ] \\
[\text{MODE } imode ] \\
\text{DIM } dim (dxyz(i), i = 1, dim) \\
;
\]

where

- **EDIT** keyword used to modify the print level *iprint*.
- **iprint** integer index used to control the printing in module VAL:. =0 for no print; =1 for minimum printing (default value); larger values of *iprint* will produce increasing amounts of output.
- **MODE** keyword to specify the flux harmonic index *imode*.
- **imode** index of the flux harmonic recovered by the VAL: module if the MONI keyword was set in module FLUD: (see Sect. 1.9.1). By default, it is assumed that the MONI keyword was not used.
- **DIM** keyword to specify the number *dim*.
- **dim** number of dimension of the geometry.
$dxyz$ mesh interval along each direction which is used to define the grid where the flux is interpolated.
2 EXAMPLES OF INPUT DATA FILES

2.1 IAEA-2D benchmark

The IAEA-2D benchmark is defined in Refs. 3, 19 and its geometry is represented in Fig. 11. Here, it is solved using a parabolic variational collocation method without mesh splitting of the elements:

![Figure 11: Description of the IAEA-2D benchmark.](image)

```
LINKED_LIST IAEA MACRO TRACK SYSTEM FLUX EDIT ;
*
IAEA := GEO: :: CAR2D 9 9
EDIT 2
  X- DIAG X+ VOID
  Y- SYME Y+ DIAG
  MIX 3 2 2 2 3 2 2 1 4
      2 2 2 2 2 1 4
      2 2 2 2 1 1 4
      2 2 2 1 4 4
      3 1 1 4 0
      1 4 4 0
      4 0 0
      0 0
      0
  MESHX 0.0 20.0 40.0 60.0 80.0 100.0 120.0 140.0 160.0 180.0
;  MACRO := MAC: ::
EDIT 2 NGRO 2 NMIX 4
READ INPUT
MIX 1
  DIFF 1.500E+00 4.0000E-01
  TOTAL 3.012E-02 8.0032E-02
  NUSIGF 0.000E+00 1.3500E-01
  H-FACTOR 0.000E+00 1.3500E-01
  SCAT 1 1 0.0 2 2 0.0 0.2E-01
MIX 2
  DIFF 1.500E+00 4.0000E-01
```
2.2 Biblis-2D benchmark

The rods-withdrawn configuration of the Biblis-2D benchmark is defined in Ref. 3 and its geometry is represented in Fig. 12. Here, it is solved using a parabolic variational collocation method without mesh splitting of the elements:
Figure 12: Description of the Biblis-2D benchmark, rods-withdrawn configuration.

MESHX 0.0 23.1226 46.2452 69.3678 92.4904 115.613 138.7356
161.8582 184.9808 208.1034

MACRO := MAC: ::
EDIT 2 NGRO 2 NMIX 8
READ INPUT
MIX 1
  DIFF 1.436000E+00 3.635000E-01
  TOTAL 2.725820E-02 7.505800E-02
  NUSIGF 5.870800E-03 9.606700E-02
  H-FACTOR 2.376800E-03 3.889400E-02
  SCAT 1 1 0.0 2 2 0.0 1.775400E-02
MIX 2
  DIFF 1.436600E+00 3.636000E-01
  TOTAL 2.729950E-02 7.843600E-02
  NUSIGF 6.190800E-03 1.035800E-01
  H-FACTOR 2.506400E-03 4.193500E-02
  SCAT 1 1 0.0 2 2 0.0 1.762100E-02
MIX 3
  DIFF 1.320000E+00 2.772000E-01
  TOTAL 2.576220E-02 7.159600E-02
  SCAT 1 1 0.0 2 2 0.0 2.310600E-02
MIX 4
  DIFF 1.438900E+00 3.638000E-01
  TOTAL 2.746400E-02 9.140800E-02
  NUSIGF 7.452700E-03 1.323600E-01
  H-FACTOR 3.017300E-03 5.358700E-02
  SCAT 1 1 0.0 2 2 0.0 1.710100E-02
MIX 5
  DIFF 1.438100E+00 3.665000E-01
  TOTAL 2.729300E-02 8.482800E-02
  NUSIGF 6.190800E-03 1.035800E-01
  H-FACTOR 2.506400E-03 4.193500E-02
  SCAT 1 1 0.0 2 2 0.0 1.729000E-02
MIX 6
DIFF 1.438500E+00 3.665000E-01
TOTAL 2.732400E-02 8.731400E-02
NUSIGF 6.428500E-03 1.091100E-01
H-FACTOR 2.602600E-03 4.417400E-02
SCAT 1 1 0.0 2 2 0.0 1.719200E-02

MIX 7
DIFF 1.438900E+00 3.679000E-01
TOTAL 2.729000E-02 8.802400E-02
NUSIGF 6.190800E-03 1.035800E-01
H-FACTOR 2.506400E-03 4.193500E-02
SCAT 1 1 0.0 2 2 0.0 1.712500E-02

MIX 8
DIFF 1.439300E+00 3.680000E-01
TOTAL 2.732100E-02 9.051000E-02
NUSIGF 6.428500E-03 1.091100E-01
H-FACTOR 2.602600E-03 4.417400E-02
SCAT 1 1 0.0 2 2 0.0 1.702700E-02

TRACK := TRIVAT: BIBLIS ::
TITLE 'BIBLIS BENCHMARK'
EDIT 5 MAXR 81 PRIM 2 ;
SYSTEM := TRIVAA: MACRO TRACK ::
EDIT 5 ;
FLUX := FLUD: SYSTEM ::
EDIT 2 ;
EDIT := OUT: FLUX ::
EDIT 2 INTG
 1 2 3 4 5 6 7 8 0
 9 10 11 12 13 14 15 0
16 17 18 19 20 21 0
22 23 24 25 26 0
27 28 29 0 0
30 31 0 0
0 0
0 0
0

END: ;
2.3 IAEA-3D benchmark

The IAEA-3D benchmark is defined in Ref. 19 and its geometry is represented in Fig. 13. Here, it is solved using a cubic mixed-dual method with mesh splitting of the second axial plane:

![IAEA-3D benchmark schematic](image_url)

Figure 13: Description of the IAEA-3D benchmark.

```
LINKED_LIST IAEA3D MACRO TRACK SYSTEM FLUX EDIT ;
*
IAEA3D := GEO: :: CAR3D 9 9 4
   EDIT 2
   X- DIAG X+ VOID
   Y- SYME Y+ DIAG
   Z- VOID Z+ VOID
   MESHX 0.0 20.0 40.0 60.0 80.0 100.0 120.0 140.0 160.0 180.0
   MESHZ 0.0 20.0 280.0 360.0 380.0
   SPLITZ 1 2 1 1
   (* PLANE NB 1 *)
   MIX 4 4 4 4 4 4 4 4
      4 4 4 4 4 4 4 4
      4 4 4 4 4 4 4 4
      4 4 4 4 4 4 4 0
      4 4 4 0
      4 0 0
```
0 0
0
(* PLANE NB 2 *)
3 2 2 2 3 2 2 1 4
2 2 2 2 2 2 1 4
2 2 2 2 2 1 1 4
2 2 2 2 1 4 4
3 1 1 4 0
1 4 4 0
4 0 0
0 0
0
(* PLANE NB 3 *)
3 2 2 3 2 2 1 4
2 2 2 2 2 2 1 4
3 2 2 2 1 1 4
2 2 2 1 4 4
3 1 1 4 0
1 4 4 0
4 0 0
0 0
0
(* PLANE NB 4 *)
5 4 4 4 5 4 4 4 4
4 4 4 4 4 4 4 4 4
5 4 4 4 4 4 4 4 4
4 4 4 4 4 4 4 4 4
5 4 4 4 0
4 4 4 0
4 0 0
0 0
0

; MACRO := MAC: ::
EDIT 2 NGRO 2 NMIX 5
READ INPUT
MIX 1
  DIFF 1.500E+00 4.0000E-01
  TOTAL 3.000E-02 8.0000E-02
  NUSIGF 0.000E+00 1.3500E-01
  H-FACTOR 0.000E+00 1.3500E-01
  SCAT 1 1 0.0 2 2 0.0 0.2E-01
MIX 2
  DIFF 1.500E+00 4.0000E-01
  TOTAL 3.000E-02 8.5000E-02
  NUSIGF 0.000E+00 1.3500E-01
  H-FACTOR 0.000E+00 1.3500E-01
  SCAT 1 1 0.0 2 2 0.0 0.2E-01
MIX 3
  DIFF 1.500E+00 4.00000E-01
  TOTAL 3.000E-02 1.30000E-01
  NUSIGF 0.000E+00 1.35000E-01
  H-FACTOR 0.000E+00 1.35000E-01
  SCAT 1 1 0.0 2 2 0.0 0.2E-01
MIX 4
**DIFF** 2.000E+00 3.0000E-01
**TOTAL** 4.000E-02 1.0000E-02
**SCAT** 1 1 0.0 2 2 0.0 0.4E-01

**MIX** 5
**DIFF** 2.000E+00 3.0000E-01
**TOTAL** 4.000E-02 5.5000E-02
**SCAT** 1 1 0.0 2 2 0.0 0.4E-01

;  
**TRACK** := **TRIVAT**: **IAEA3D** ::
**TITLE** 'TEST IAEA 3D'
**EDIT** 5 **MAXR** 405 **DUAL** 3 1 ;
**SYSTEM** := **TRIVAA**: **MACRO** **TRACK** ::
**EDIT** 5 ;
**FLUX** := **FLUD**: **SYSTEM** ::
**EDIT** 2 ;
**EDIT** := **OUT**: **FLUX** ::
**EDIT** 2 **INTG**

(* PLANE NB 1 *)
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0

(* PLANE NB 2 *)
1 2 3 4 5 6 7 8 0
9 10 11 12 13 14 15 0
16 17 18 19 20 21 0
22 23 24 25 0 0
26 27 28 0 0
29 0 0 0
0 0 0
0 0
0

(* PLANE NB 3 *)
30 31 32 33 34 35 36 37 0
38 39 40 41 42 43 44 0
45 46 47 48 49 50 0
51 52 53 54 0 0
55 56 57 0 0
58 0 0 0
0 0 0
0 0
0

(* PLANE NB 4 *)
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0

(* PLANE NB 5 *)
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
2.4 S30 hexagonal benchmark in 2-D

The S30 hexagonal benchmark in 2-D is defined in Ref. 15. Its geometry is represented in Fig. 14. Here, it is solved using a mesh centered finite difference method without mesh splitting of the hexagonal elements:

![Diagram of S30 hexagonal benchmark](image)

Figure 14: Description of the S30 hexagonal benchmark.

```plaintext
LINKED_LIST HEX MACRO TRACK SYSTEM FLUX EDIT ;
*
HEX := GEO: :: HEX 6
  EDIT 2
  HBC S30 ZERO
  SIDE 13.044
  SPLITH 0
  MIX
  1
  2
  2 2
  3 3

MACRO := MAC: ::
EDIT 2 NGRO 2 NMIX 3
READ INPUT
MIX 1
  DIFF 1.5E+00 4.00E-01
  TOTAL 3.0E-02 1.30E-01
  NUSIGF 0.0E+00 1.35E-01
  H-FACTOR 0.0E+00 1.35E-01
  SCAT 1 1 0.0 2 2 0.0 0.2E-01
MIX 2
  DIFF 1.5E+00 4.00E-01
  TOTAL 3.0E-02 8.50E-02
  NUSIGF 0.0E+00 1.35E-01
  H-FACTOR 0.0E+00 1.35E-01
  SCAT 1 1 0.0 2 2 0.0 0.2E-01
MIX 3
  DIFF 2.0E+00 3.0E-01
  TOTAL 4.0E-02 1.0E-02
```
2.5 LMW benchmark in 2-D

The LMW benchmark in 2-D is a space-time kinetics problem introduced by Greenman\cite{20} and used by Monier\cite{14}. Its geometry is represented in Fig. 15. Here, it is solved using a parabolic nodal collocation method with $2 \times 2$ mesh splitting of each element. A reactivity transient is induced by the rapid withdrawal of the control rod in material mixture 6. The control rod is removed in 26.7 s, causing a negative ramp variation in total cross section.

![Figure 15: Description of the LMW benchmark in 2-D.](image-url)
END; 
REAL fnorm sigt1 sigt2 ; 
REAL TIME := 0.0 ; 
PROCEDURE assertS assertS2 ; *
LMW := GEO: :: CAR2D 6 6
  X- REFL X+ ZERO
  Y- REFL Y+ ZERO
  MIX 1 1 1 2 3 4
      1 1 1 1 3 4
      1 1 5 1 3 4
      6 1 1 3 3 4
      3 3 3 3 4 4
      4 4 4 4 4 0
MESHX 0.0 10. 30. 50. 70. 90. 110.
MESHY 0.0 10. 30. 50. 70. 90. 110.
SPLITX 2 2 2 2 2
SPLITY 2 2 2 2 2
;
MACRO1 := MAC: ::
EDIT 0 NGRO 2 NMIX 6
READ INPUT
MIX 1
  DIFF 1.423910E+00 3.563060E-01
  TOTAL 2.795756E-02 8.766216E-02
  NUSIGF 6.477691E-03 1.127328E-01
  H-FACTOR 2.591070E-03 4.509310E-02
  SCAT 1 0.0 2 2 0.0 0.175555E-01
  OVERV 0.800E-07 4.000E-06
MIX 2
  DIFF 1.423910E+00 3.563060E-01
  TOTAL 2.850756E-02 9.146219E-02
  NUSIGF 6.477691E-03 1.127328E-01
  H-FACTOR 2.591070E-03 4.509310E-02
  SCAT 1 1 0.0 2 2 0.0 0.175555E-01
  OVERV 0.800E-07 4.000E-06
MIX 3
  DIFF 1.425610E+00 3.505740E-01
  TOTAL 2.817031E-02 9.925634E-02
  NUSIGF 7.503282E-03 1.378004E-01
  H-FACTOR 3.001310E-03 5.512106E-02
  SCAT 1 1 0.0 2 2 0.0 0.171777E-01
  OVERV 0.800E-07 4.000E-06
MIX 4
  DIFF 1.634220E+00 2.640020E-01
  TOTAL 3.025750E-02 4.936351E-02
  SCAT 1 1 0.0 2 2 0.0 0.275969E-01
  OVERV 0.800E-07 4.000E-06
MIX 5
  DIFF 1.423910E+00 3.563060E-01
  TOTAL 2.795756E-02 8.766216E-02
  NUSIGF 6.477691E-03 1.127328E-01
  H-FACTOR 2.591070E-03 4.509310E-02
  SCAT 1 1 0.0 2 2 0.0 0.175555E-01
  OVERV 0.800E-07 4.000E-06
MIX 6
DIFF 1.423910E+00 3.563060E-01
TOTAL 2.850756E-02 9.146217E-02
NUSIGF 6.477691E-03 1.127328E-01
H-FACTOR 2.591070E-03 4.509310E-02
SCAT 1 1 0.0 2 2 0.0 0.175555E-01
OVERV 0.800E-07 4.000E-06

; TRACK := TRIVAT: LMW ::
  TITLE 'LMW 2-D BENCHMARK'
  EDIT 1 MAXR 144 MCFD 2 ;
SYSTEM1 := TRIVAA: MACRO1 TRACK ::
  EDIT 1 UNIT ;
FLUX := FLUD: SYSTEM1 TRACK ::
  EDIT 1 EXTE 5.0E-7 ;
assertS FLUX :: 'K-EFFECTIVE' 1 1.014803 ;
*----
* Crank-Nicholson space-time kinetics
*----
EVALUATE TIME := 0.0 ;
KINET := INIKIN: MACRO1 TRACK SYSTEM1 FLUX :: EDIT 1
  NDEL 6
  BETA 0.000247 0.0013845 0.001222 0.0026455 0.000832 0.000169
  LAMBDA 0.0127 0.0317 0.115 0.311 1.40 3.87
  CHID 1.0 1.0 1.0 1.0 1.0 1.0
       0.0 0.0 0.0 0.0 0.0 0.0
  NORM POWER-INI 1.0E4 ;
EVALUATE sigt1 := 2.850756E-02 ;
EVALUATE sigt2 := 9.146217E-02 ;
WHILE TIME 26.7 <= DO
  EVALUATE sigt1 := sigt1 5.5E-4 0.1 26.7 / * - ;
  EVALUATE sigt2 := sigt2 3.8E-3 0.1 26.7 / * - ;
MACRO2 := MAC: MACRO1 ::
  EDIT 0
  READ INPUT
  MIX 6
  TOTAL <<sigt1>> <<sigt2>>
; SYSTEM2 := TRIVAA: MACRO2 TRACK ::
  EDIT 1 UNIT ;
KINET := KINSOL: KINET MACRO2 TRACK SYSTEM2 MACRO1 SYSTEM1 ::
  EDIT 5 DELTA 0.1
  SCHEME FLUX CRANK PREC CRANK EXTE 1.0E-6 ;
GREP: KINET :: GETVAL 'TOTAL-TIME' 1 >>TIME<< ;
ECHO "TIME=" TIME "S" "sigt=" sigt1 sigt2 ;
IF TIME 1.0 - ABS 1.0E-3 < THEN
  assertS2 KINET :: 'CTRL-FLUX' 1 1.986270E+02 ;
  assertS2 KINET :: 'CTRL-PREC' 1 1.095509E-01 ;
  assertS2 KINET :: 'E-POW' 1 1.008753E+04 ;
ELSEIF TIME 5.0 - ABS 1.0E-3 < THEN
  assertS2 KINET :: 'CTRL-FLUX' 1 2.090369E+02 ;
  assertS2 KINET :: 'CTRL-PREC' 1 1.097266E-01 ;
  assertS2 KINET :: 'E-POW' 1 1.063990E+04 ;
ELSEIF TIME 10.0 - ABS 1.0E-3 < THEN
  assertS2 KINET :: 'CTRL-FLUX' 1 2.305455E+02 ;
assertS2 KINET :: 'CTRL-PREC' 1 1.104699E-01 ;
assertS2 KINET :: 'E-POW' 1 1.176902E+04 ;
ELSEIF TIME 15.0 - ABS 1.0E-3 < THEN
  assertS2 KINET :: 'CTRL-FLUX' 1 2.641221E+02 ;
  assertS2 KINET :: 'CTRL-PREC' 1 1.121002E-01 ;
  assertS2 KINET :: 'E-POW' 1 1.352433E+04 ;
ENDIF ;
MACRO1 SYSTEM1 := DELETE: MACRO1 SYSTEM1 ;
MACRO1 := MACRO2 ;
SYSTEM1 := SYSTEM2 ;
MACRO2 SYSTEM2 := DELETE: MACRO2 SYSTEM2 ;
ENDWHILE ;
ECHO "test lmw2D completed" ;
References


Index

\[ i_{src1}, 32, 34 \]
\[ i_{src2}, 32 \]
\[ i_{reg}, 10 \]
\[ ::, 2, 13, 18, 21, 24, 25, 27, 30, 32, 35, 37, 39, 43, 46 \]
\[ ::=, 2, 13, 18, 21, 24, 25, 27, 30, 32, 35, 39, 43, 46 \]
\[ ;, 1 \]
\[ ACCE, 27, 28, 32, 33, 44, 45 \]
\[ ACYL, 4, 5 \]
\[ ADI, 21, 23, 27, 28, 32, 33, 44, 45 \]
\[ ADJ, 27, 29–31, 44, 45 \]
\[ ALBE, 4, 5 \]
\[ albedo, 4, 5 \]
\[ albedp, 13, 14 \]
\[ ALBP, 13, 14 \]
\[ ALL, 9, 32, 34 \]
\[ ANG, 4, 8 \]
\[ ang, 4, 8 \]
\[ ANIS, 13, 14 \]
\[ BETA, 39, 40 \]
\[ beta, 39, 40 \]
\[ (BIVACA:), 24 \]
\[ BIVACA:, 24 \]
\[ (bivaca_data), 24 \]
\[ (BIVACT:), 18 \]
\[ BIVACT:, 18 \]
\[ (bivact_data), 18 \]
\[ bsz, 27, 28 \]
\[ CAR1D, 3, 11 \]
\[ CAR2D, 3, 5, 8, 11 \]
\[ CAR3D, 3, 5, 8, 10–12 \]
\[ CHDL, 15, 16 \]
\[ CHI, 15, 16 \]
\[ CHID, 39, 40 \]
\[ chid, 39, 40 \]
\[ COMPLETE, 4, 5, 9 \]
\[ COND, 35, 36 \]
\[ CRANK, 44, 45 \]
\[ CROWN, 9 \]
\[ CYLI, 4, 5 \]
\[ DELP, 13, 14 \]
\[ DELTA, 44 \]
\[ delta, 44 \]
\[ (DELTa:), 30 \]
\[ DELTA:, 30 \]
\[ (delta_data), 30 \]
\[ DERI, 25 \]
\[ (descBC), 3, 4 \]
\[ (descMC), 3, 4, 9 \]
\[ (descPOS), 3, 4, 10 \]
\[ (descval), 46 \]
\[ DIAG, 4, 5, 8, 10 \]
\[ DIFF, 15, 16, 18, 19, 21, 22 \]
\[ diff, 15, 16 \]
\[ DIFFX, 15, 16 \]
\[ DIFFY, 15, 16 \]
\[ DIFFZ, 15, 16 \]
\[ DIM, 46 \]
\[ dim, 46 \]
\[ DIRE, 35, 36 \]
\[ DMACRO, 25, 26 \]
\[ DOLD, 13, 15 \]
\[ DSYST, 30 \]
\[ DUAL, 18, 19, 21, 22 \]
\[ dxyz, 46, 47 \]
\[ EDIT, 3, 4, 13, 18, 21, 24, 25, 27, 30–33, 35, 39, 40, 44, 46 \]
\[ END: ;, 1 \]
\[ EPSG, 27, 28 \]
\[ epsinsr, 27, 28 \]
\[ epsout, 27, 28, 32, 33, 44, 45 \]
\[ epsthr, 27, 28, 32, 33, 44, 45 \]
\[ (ERROR:), 37 \]
\[ ERROR:, 37 \]
\[ EXPLICIT, 32–34 \]
\[ EXPON, 44, 45 \]
\[ EXTE, 27, 28, 32, 33, 44, 45 \]
\[ FIXE, 15, 16 \]
\[ (FLUD:), 27 \]
\[ FLUD:, 27 \]
\[ (fluid_data), 27 \]
\[ FLUX., 27, 29, 35, 39, 40 \]
\[ FLUX, 44 \]
\[ FLUX0, 30, 32 \]
\[ FLUX_GPT, 32, 33 \]
\[ FLUXUNK, 46 \]
\[ fnorm, 40 \]
\[ FROM-TO, 32, 34 \]
\[ (GEO:), 2 \]
\[ GEO:, 2 \]
\[ GEO: ;, 2 \]
\[ (geo_data1), 2, 3 \]
\[ (geo_data2), 2, 3 \]
\[ GEOM, 18, 21, 35 \]
\[ GEOM1, 2 \]
\[ GEOM2, 2 \]
\[ GMRES, 32, 33 \]
\[ GPT, 30, 32, 34 \]
\[ (GPTFLU:), 32 \]
\[ GPTFLU:, 32 \]
(gptflu_data), 32
H–FACTOR, 15, 16
HBC. 4
HEX, 3, 11
HEXZ, 3, 11, 12
hfact, 16
hname, 37, 38
HOMOGE. 3
HREA, 37, 38
i, 9
ic, 9
icl1, 27, 28, 32, 33, 44, 45
icl2, 27, 28, 32, 33, 44, 45
icode, 4, 5
icol, 18, 19, 21, 22
icond, 35, 36
ielem, 18, 19, 21–23
IFLU, 46
ihom, 35, 36
ilastg, 15, 17
imix, 9
imode, 35, 46
IMPLICIT, 44, 45
IMPLICITIT, 32, 34
imprv, 21, 23
IN, 35, 36
(INIKIN::), 39
INIKIN:, 24, 26, 39
(inikin_data), 39, 40
INPUT, 13
INTERPFLUX, 46
INTG, 35, 36
iplan, 9
iplan1, 9
iprint, 3, 4, 13, 18, 21, 22, 24, 25, 27, 30–33, 35, 39, 40, 44, 46
IRAM, 27, 28
iscat, 18, 19, 21, 22
iseg, 21, 23
ispltth, 10, 11
isptl, 10, 11
isplitr, 10, 11
isplitx, 10, 11
isplty, 10, 11
ispltz, 10, 11
istep, 13, 14
jmix, 9
KINET, 39, 43
(KINSOL::), 43
KINSOL::, 24, 26, 43
(kinsol_data), 43, 44
korg, 27, 28
LAMDA, 39, 40
lambda, 39, 40
lc, 9
lh, 3, 4, 11, 12
LINKED_LIST, 1
lmod, 27, 29
lp, 9
lr, 3, 4, 10
LUMP, 21, 22
lx, 3, 10
ly, 3, 4, 10
lz, 3, 4, 10
(MAC:), 13
MAC:, 5, 13
(mac_data), 13
MACLIB, 14
MACR1, 13, 14
MACR2, 13
MACRO, 24–27, 35, 39, 43, 44
MACRO1, 37, 38
MACRO2, 35, 37, 38
MACRO3, 43, 44
(macxs), 13–15
mature, 15, 16
MAX, 40
maxout, 27, 28, 32, 33, 44, 45
maxpts, 11, 12, 18, 19, 21, 22
MAXR, 18, 19, 21, 22
maxthr, 27, 28, 32, 33, 44, 45
MCFD, 18, 19, 21, 22
MESHX, 10
MESHEY, 10
MESHZ, 10
MIX, 9, 15, 35, 36
MODE, 35, 46
MODULE, 1
MONI, 27, 29
n, 18, 19, 21, 22
nadi, 21, 23, 27, 28, 32, 33, 44, 45
naih, 13, 14
NAME1, 1
NAME2, 1
NAME3, 1
NAME4, 1
NAME5, 1
naniso, 13–15, 17
nblocat, 15, 17
NDEL, 39, 40
ndel, 13–15
ndg, 13, 14, 39, 40
ngcon, 36
NGRO, 13, 14
ngroup, 13–15, 17, 36
NGRP, 39, 40
ngrp, 39, 40
NIFI, 13, 14
nifiss, 13–15
NMIX, 13, 14
nmix, 9, 13–15
NONE, 35, 36
NORM, 40
nrads, 4, 8
NREG, 37, 38
nreg, 37, 38
nstart, 27, 28
nstart, 32, 33
NTOTO, 15
NTOTI, 15
NUSIGD, 15, 16
NUSIGF, 15, 16
avd, 18, 19, 21, 22
OLD, 13, 14
(OUT:), 35
OUT:, 35
(out_data), 35
OVEL, 25, 26
OVERV, 15, 16
overv, 15, 16
PERT, 25, 26
PICK, 44, 45
PLANE, 9
PN, 18, 19
cpower, 35, 36, 40
POWER-INI, 40
cpower_out, 44, 45
PWR, 35, 36
PREC, 44
PRIM, 18, 19, 21, 22
PRD, 35, 36
PRTV, 21, 23
R+, 4
R120, 4, 5
R180, 4, 5
RADIUS, 10
RADS, 4, 8
RAND, 27, 29
READ, 13, 14
REFL, 4, 5
RELAX, 27, 29
relax, 27, 29
rrad, 4, 8
rrr, 10
S30, 4, 5
S90, 4, 5
SA180, 4, 5
SA60, 4, 5
SAME, 9
SB180, 4, 5
SB60, 4, 5
SCAT, 15, 16, 18, 19, 21, 22
scat, 15
SCHEME, 44
SEQ_ASCII, 1
SEQ_BINARY, 1
SIDE, 10
sidhex, 10
SKIP, 25
snumb, 35, 36
SOUR, 35, 36
(specif), 1
SPHERE, 3, 10, 11
SPLITL, 10, 11
SPLITR, 10, 11
SPLITX, 10
SPLITY, 10
SPLITZ, 10
SPN, 18, 19, 21, 22
STEP, 13, 14
SYME, 4, 5
SYST, 24–27, 32, 39, 43
SYST0, 30
SYST0, 43, 44
TEXP, 44
THER, 27, 28, 32, 33, 44, 45
THETA, 44, 45
theta, 45
TITL, 18, 19, 21, 22
TITLE, 18, 19, 21, 22
TOTAL, 15
TRACK, 18, 21, 24, 25, 27, 29, 30, 32, 35, 39, 43–45
TRACKING, 46
TRAN, 4, 5, 8
TRANC, 15
(trip2), 13, 15
(triv2), 13, 14
(TRIVAA:), 25
TRIVAA:, 25
(trivaa_data), 25
(TRIVAC), 1
(TRIVAT:), 21
TRIVAT:, 21
(trivat_data), 21
TRKNNAM, 46
tttfix, 44, 45
ttprc, 44, 45
TUBE, 3, 10, 11
TUBEZ, 3, 10, 11
UNIT, 24–26
**UPTO, 9**

**(VAL:, 46**

**VAL:, 46**

**VAR1: 27, 28, 32, 33, 44, 45**

**VECT, 21, 23**

**VOID, 4, 9, 18, 19, 21, 22**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X+</td>
<td>4, 8–10</td>
</tr>
<tr>
<td>X–</td>
<td>4, 8–10</td>
</tr>
<tr>
<td>xdiffx</td>
<td>15, 16</td>
</tr>
<tr>
<td>xdiffy</td>
<td>15, 16</td>
</tr>
<tr>
<td>xdiffz</td>
<td>15, 16</td>
</tr>
<tr>
<td>xhfact</td>
<td>15</td>
</tr>
<tr>
<td>xrad</td>
<td>4, 8</td>
</tr>
<tr>
<td>xschi</td>
<td>15, 16</td>
</tr>
<tr>
<td>xschid</td>
<td>15, 16</td>
</tr>
<tr>
<td>xsfixe</td>
<td>15, 16</td>
</tr>
<tr>
<td>XSM_FILE</td>
<td>1</td>
</tr>
<tr>
<td>xsscat</td>
<td>17</td>
</tr>
<tr>
<td>xssig1</td>
<td>15</td>
</tr>
<tr>
<td>xssigd</td>
<td>15, 16</td>
</tr>
<tr>
<td>xssigf</td>
<td>15, 16</td>
</tr>
<tr>
<td>xssigt</td>
<td>15</td>
</tr>
<tr>
<td>xsstera</td>
<td>15, 16</td>
</tr>
<tr>
<td>xxx</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y+</td>
<td>4, 8–10</td>
</tr>
<tr>
<td>Y–</td>
<td>4, 8–10</td>
</tr>
<tr>
<td>yyy</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z+</td>
<td>4, 8–10</td>
</tr>
<tr>
<td>Z–</td>
<td>4, 8–10</td>
</tr>
<tr>
<td>ZERO</td>
<td>4, 5</td>
</tr>
<tr>
<td>zzz</td>
<td>10</td>
</tr>
</tbody>
</table>