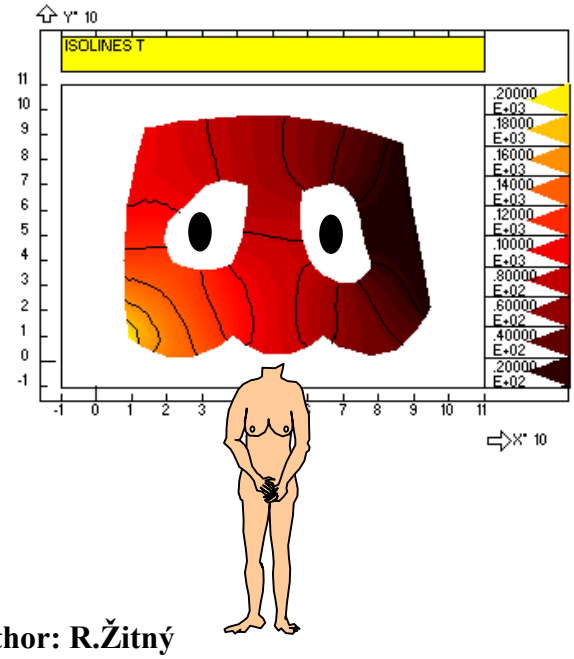
FEMINA

(Finite Element Method for thermal, flow, stress and residence time distribution analysis)

Part I-3

User manual COMMANDS SUMMARY



Author: R.Žitný

March 2005

3. COMMANDS SUMMARY

3.1. How to start and end a job

NEWPROB (N)[1]	name	opens files *name*.SES, *name*.OUT, *name*.DBG and initialises arrays
INIFEM		initialisation of FEM arrays (it has a sense when repeating a FEM task)
CMD		reread command file $FEMINA.CMD
F[2] (FILE)	filename	batch processing of a command file (GEO postfix is default)
M (MAC)	filename	similar as FILE, with the possibility to define missing parameters interactively
UNDO	no. of backsteps	undo specified number of steps
EXIT		terminates execution of FEMINA and closes all opened files.

3.2. Entities

Definition of a specific FEM problem includes definition of entities describing geometry, finite elements, node points and material properties. Similar names of entities (e.g. PT) and derived operations (e.g. PTLIST) are used also in the COSMOS program:

PT	keypoint (as basic geometry entity, it is not a node)
CR	curve (defined by the list of 2 or 3 points PT)
CT	contour (a list of curves)
SF	surface (quadrilateral determined by 4 or 8 points, or triangle determined by 3 or 6 PTs)
VL	volume (hexahedron determined by the list of 8 or 20 points)
EL	element (defined by the list of nodes)
ND	node point.

Degrees of freedom (DOF) are assigned to every node and their names can be used as command parameters:

1	2	3	4	5	6	7	8	9	10	11	12
TEMP	**UX**	**UY**	**UZ**	**RX**	**RY**	**RZ**	**VOLT**	**VX**	**VY**	**VZ**	**PRES**
temperature	displacements			rotations			voltage	velocities			pressure

13	14	15	16	17	18	19	20	21	22	23	24
OMG	**PS**	**PSX**	**PSY**	**PSXX**	**PSYY**	**PSXY**	**CN**	**CD**	**CA**	**KT**	**EPS**
vorticity	ψ	$\psi_{,x}$	$\psi_{,y}$	$\psi_{,xx}$	$\psi_{,yy}$	$\psi_{,xy}$	concentrations			k	ε

Nodal parameters are assigned to every *node,* while information about materials, dimensions (if they cannot be deduced from theirs coordinates), etc., are assigned to *elements*. All the information belongs to one of three groups and every group is described by a table: the table of material parameters (rows represent a material, and columns represent specific property, for example the first column is thermal conductivity), then the table real constants (e.g., the element width is in the first column, its diameter in the second column, inner pressure in the third,…), and finally the table which specifies the element type and computation algorithm (e.g. whether the mass matrix should be calculated, how many points should be used in the Gauss integration etc.).

EGROUP – element characteristics (integer values)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
element type	Static/ transient	Gauss points	cartesian / cylindr.	stress/ strain	solution method	Hydr. charact.	Heat source	Heat sink	C_N source	C_N sink	C_D source	C_D sink	C_A source	C_A sink	parallel pipes
ENAME	TRANS	GAUSS	AXIS	STRESS	TYPE	HYDR	HEAT	HSINK							

0-cartesian,1-cylindrical, 2-spatial extension of 1D elements

[1] Words in parenthesis are synonyms – usually abbreviated command names.
[2] F – makes use dialog window, unlike FILE which is restricted to console. The same difference is between M and MAC.

1-PIPE2D,3-TRUSS2D,7-CSTR,8-PUMP,10-SHELLAX,11-HEXC,12-PLANE2D,12-FLOW2D,13-FLOW3D,15-USER,17-SOLID,18-SPHERE,19-CYLINDER

1	2	3	4	5	6	7	8	9	10	11	18	19	20
thermal conduct.	thermal capacity	density	electric conduct.	elasticity modulus	Poisson. const.	dynam. viscosity	volumetr. expans.	difusion coeff.	Enthalpy of format	frequen. factor	linear expans.	Melting point	Latent heat
W/m/K	J/kg/K	kg/m^3	S/m	Pa	-	Pa.s	1/K	m^2/s	J/mol		1/K	^{0}C	J/kg
KX	CP	DENS	KAPPA	EX	MI	VISC	BETA	DN	HN	AN	ALEX	TM	HM

MPROP-material properties (values D,H,A for other components are in columns 12 to 17)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
wall width	equival. diam.	external pressure	heat transfer	external temp.	cross-section	channel perim.	inner volume	roughness	el. field intens.	mom. inertia	ζ	permeability	k_N	c_{eN}	k_D	c_{eD}	k_A	c_{eA}
m	m	Pa	W/m^2/K	K	m^2	m	m^3	-	V/m	m^4	-	m^2						
HH	DD	PE	ALPHA	TE	AREA	PERIM	VOL	RA	DUDY	JZ	ZETA	PERM						

RCONST - element charakteristics (real values)

Identifiers of system vectors, e.g.
HH(2)=0.1 assigns thickness in group 2.

TC entity Time Courses consists of up to 9 time courses represented by a table MOP – matrix of observation points, the first column is time which is common for the following columns 2,3,…,10. This entity is used either by finite elements for time-history records of selected nodal parameters or integrals in transient problems and of course also by RTD models for records of responses, or for experimental data. The MOP serves for regression analysis (independent and dependent variables, standard deviations of data points) and generally for any comparison between model prediction and measured or calculated data, i.e. for optimisation algorithms.

3.3. List entities (all outputs are written into the file *.DBG and displayed in the LIST window)

U	n-lines up	scroll the LIST window a specified number of lines up
D	n-lines down	scroll the LIST window a specified number of lines down
PU	scroll the LIST window one page up	
PD	scroll the LIST window one page down	
REM	writes a comment into the output file *.DBG	
PTLIST (PTL)	first point,last point,increment	list of point coordinates
CRLIST (CRL)	first curve,last curve,increment	list of points creating curves
CTLIST (CTL)	first contour,last ccontour,increment	list contours
SFLIST (SFL)	first surface,last surface,increment	list of points creating surfaces
VLLIST (VLL)	first volume,last volume,increment	list of points creating volumes
NDLIST (NDL)	first node,last node,increment,DOF	nodal coordinates and parameters
NFLIST (NFL)	first node,last node,increment, DOF	nodes with *status*≠0 (bound.conditions,load)
ELIST (EL)	first element,last element,increment	connectivity matrix and group numbers
EPLIST (EPL)	first element,last element,increment,name	results of postprocessing are in the matrix of element parameters EPAR. A specific parameter is identified by *name* (e.g. ERE,EQ,…).
MPLIST (MPL)	group of material properties	material properties
RCLIST (RCL)	group of real constants	other element characteristics (REAL)
EGLIST (EGL)	group of elements	element parameters (INTEGER)
CLIST (CL)		list of options (upwind, fouling, buoyancy,…)
CURLIST (CURL)		list of all defined tables
FUNLIST (FUNL)		list of all defined functions
DOFLIST (RANGE)	‚Zone 1-BC,2-res,3-IC'	list of min/max values of all DOF (BC-boundary conditions, RES-computation result, IC-initial conditions)
VARLIST (VARL)	'List of variables=0, with comments=1'	lists all variables of command interpreter
LOC	'Expression'	lists information about a variable (or an expression)

MODLIST (MODL) lists information about the actual model of ordinary differential equations

PALIST (PL) RTD model parameters will be reported in the LIST window

TCLIST (TCL) 'first row','format' matrix of observation points is displayed in the model window.

3.4. Data export and import

WRITE (W) 'Write 1-Nod,2-Ele,3-Groups,4-Epar,5-PCS,6-Dat,7-TC'

writes selected kind of data to files *name*.NOD (nodes and node parameters), *name*.ELE (connectivity matrix), *name*.GRP (properties of elements and material parameters), *name*.EPA (parameters of elements as a result of postprocessing), *name*.PCS (geometry: Points, Curves, Surfaces), *name*.DAT (matrix of observation points), *name*.TC (part of "observation matrix", writing only one selected time course).

Form of files NOD – nodes

```
test
    441      1 (Number of nodes, DOF-active)
 Node       X          Y          Z     Kind MPU    IPU JPU TEMP    IPU JPU UX
      1 -.103E-02 -.630E-02  .000E+00  1  17      -1  1  .000E+00  0  2  .000E+0
…
    441  .101E+01  .997E+00  .000E+00  1  17      -2  1  .847E+02  0  2  .000E+0
```

Form of files ELE – connectivity matrix

```
 test
  400 (Number of elements)
 Elem. Eg  Mp  Rc  Nue   N1    N2    N3    N4    N5    N6    N7    N8
    1   1   1   1   4    -1     2    23    22
…
  400   1   1   1   4  -419  -420  -441  -440
```

Form of files GRP – EGROUP, RCONST, MPROP

```
Test          UVP
    1    1    1 (Number of E-groups, RConst, MP)
 E-group   Elem TRAN GAUS AXIS STRE TYPE HYDR H-so H-si Cn-s ….
       1    12    0    4    0    0    1    0   10    0    0

 RC-group       HH[m      DD[m      PE p      ALPH      TE        AREA ……
       1    .100E+01  .500E-01  .000E+00  .000E+00  .200E+03  .000E+00

 MP-group   KX-heat   CP-heat   DENSity   KAPPA-el  EX-Young  MI-Poiss  VISCosit
       1    .600E+00  .420E+04  .998E+03  .400E-01  .210E+12  .280E+00  .100E-02

 Index___RPN___Function
   -02   015 : 50*(1+SIN(1E-6*TIME))
    01   018 : 100*(1+SIN(10*XX*YY))
```

Form of files EPA (element parameters – results of postprocessing)

```
16 -auxilliary parameters of elements

 Index  EDISSip    ELAVolum   EIIinvar ……   EPOWER
     1   .000E+00   .000E+00   .000E+00   .669E+02  -.963E+01
     2   .000E+00   .000E+00   .000E+00   .953E+02  -.422E+01
```

Form of files PCS (geometrical entities: Points, Curves, Surfaces and Volumes)

```
Test

    21 NPT (No. of points)
 Index      X          Y          Z
     1   .000E+00   .000E+00   .000E+00
     2   .500E+00   .000E+00   .000E+00

    12 NCR (No. of curves)
 Index  NPTs PT1   PT2   PT3
    1    3    14    16    15
    2    3    16    18    17

     4 NSF (No. of surfaces)
 Index  NPTs PT1   PT2   PT3   PT4   PT5   PT6   PT7   PT8
    1    8     1     3    20    18     2    12    19     9
    2    8     3     5    14    20     4    11    21    12

     0 NVL (No. of volumes)
 Index  NPTs PT1   PT2   PT3   PT4  PT5   PT6   PT7   PT8   PT9  PT10  PT11 …
```

Form of files DAT – matrix of observation points

```
   EMPTY     XINPUT    YOUTPU    EMPTY     EMPTY     EMPTY     EMPTY     EMPTY     EMPTY     EMPTY
       0         4         5         0         0         0         0         0         0         0
   1  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00
```

…

```
   20  .190E+01  .000E+00  .148E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00  .000E+00
```

READ (R) 'Read 1-Nod,2-Ele,3-Groups,4-Epar,5-PCS,6-Dat,7-txt,8-txt,9-COSMOS,10,11-Fluent'

This command reads data in the format used by the WRITE command. This concerns matrix of nodes[3], elements, property groups and element parameters, and the format of observation matrix as well. Options 7 and 8 have special meaning. They import data .txt in a more or less arbitrary format into the matrix of observation points as either separate columns (option 7) or even with interpolation in a table t_i,c_i (option 8 - conversion of nonequdistant data with time values which do not correspond to the time basis of the observation matrix). The option 9 imports the finite element grid created by COSMOS program (nodes and connectivity matrix), option 10 performs the same from FLUENT (nodes and connectivity matrix exported from FLUENT in the neutral PATRAN format), option 11 reads nodal parameters exported from FLUENT (*.RES file).

Remark: Parameters of the READ command depend upon the first parameter – option. The second parameter is always a file name, but for options 7,8,11 additional parameters must be supplied:

Read several succesive columns from file to MOP

READ 7, *file, first column in file, corresponding column in MOP, no.of transferred columns, row of header, first row of data in the file*

Calculate one column in MOP by interpolating data t_i, c_i in a data file

READ 8, *file, index of column time t_i in data, index of column value c_i in the file, column of MOP of transformed values (recalculated according to time points defined in the first column of MOP – quadratic interpolation is used), first row of data in file*

Read nodal values from PATRAN file (*.res) – nodes and elements had to be defined by READ 10

READ 11, *file, DOF (name of nodal variable), zone (1-boundary condition, 2-results, 3-initial conditions)*

READTS (RTS) 'file-template',time step, DOF, zone

This command reads nodal values prepared in a file (PATRAN RES format is the same as in the previous READ command – option 11). A difference is only in the way, how the file name is specified: this command is used for consecutive reading of several time steps in a loop with the loop count (a variable) forming a part of filename, for example CN_____1.res, CN_____1.res, ... CN__2500.res. The first parameter is a basic file name (max.12 characters including .res), the second parameter is an integer which substitutes characters preceding dot (.) in the file-template. The parameters DOF and zone determine where to put the nodal values from the file.

Remark: Reading nodal values from FLUENT (READ option 11, or READTS) is possible only if the nodal coordinates as well as the connectivity matrix have been defined – it means read from PATRAN neutral file (using READ option 10).

UNIFIL 'file (without postfix)','template.RES',no.of time steps, time step

Creates file.OUT by joining RES-files exported from FLUENT. Example: UNIFIL F,C000.RES,30,0.15 creates file F.OUT from files C001.RES, C002.RES,…, C030.RES. The file F.OUT has the same format as results of solution obtained by FEMINA (calculated nodal values at several time steps).

FEMINA can prepare also batch-files for FLUENT. This is rather specific application aimed at simulation of experiment with injection of tracer at inlet of a continuous apparatus and calculation of the tracer distribution at several time steps after injection. This calculation is carried out by FLUENT using a model of apparatus prepared for example by GAMBIT preprocesor and using a batch file consisting of FLUENT's commands describing individual time steps of simulation, including the time steps when injection of tracer begins and when stops, how and where results of selected time steps are to saved and so on. And just this batch file can be created by FEMINA using command

FLUENT 'case file', time step, time step of injection start, injection ends, save results after each n-time steps, no.of iterations in each time step

[3] NOD-file can be prepared also in the simplified format: first row no.of nodes and following rows x,y,z,nodal value.

The following command ROM is most frequently used for reading time curves (responses) from a file into the matrix of observation points. It is possible do read several columns simultaneously and transfer them into arbitrary columns of MOP. Only in the case that the source data are to be transformed (interpolated) during reading the command READ 8 is to be used.

REOMAT (ROM) Dialog window for reading data in quite general format into observation matrix MOP.

All necessary information about geometry, properties, user defined functions, elements, nodes and nodal parameters as well as time records in the matrix of observation points form a FEMINA database. The following commands read or write a full or an abbreviated form of this database

READBI (RB) name reads the database from the file *name*.BIN (binary form)

WRITEB (WB) name,type writes the database into the file *name*.BIN. Parameter type =0 corresponds to writing the reduced database (some temporary values are omitted), for type=1 the full database is written (including unused components).

The following operation BOX does not write results but creates a direct access file $RUNBOX.BIN, which contains information about a correspondence of arbitrary coordinates *x,y,z* and elements (simply said, an element index is assigned to an arbitrary point *x,y,z* which is located inside the element, or the value –999 is assigned if the point is located outside the finite element mesh). This file is necessary for the following operations as calculation of integral of solutions, method of characteristics, processing responses of collimated detectors etc. Let us remark here that even one-dimensional elements as, for example, PIPE are considered as three-dimensional objects, so the pipe is not a line but a cylinder with diameter given by the corresponding parameter in the RCONST zone.

BOX size

The parameter size (default value 0.3) determines relative dimensions of elementary cubes which covers the whole modelled area. Value of 1 corresponds to size of the smallest element used. The lower is the size the more accurate will be operations requiring interpolation of solution. You can check results using the command BOXTES *x,y,z,zone,DOF*.

3.5. Plot entities and Drawing in window MODEL

PTPLOT (PTP) first point,last point,increment (PTPG the same but plotting to window VIEW)
CRPLOT (CRP) first curve,last curve,increment (CRPG the same but plotting to VIEW)
CTPLOT (CTP) first contour,last contour,increment
SFPLOT (SFP) first surface,last surface,increment
VLPLOT (VLP) first volume,last volume,increment
NDPLOT (NDP) first node,last node,increment
NFPLOT (NFP) first node,last node,increment, 'DOF:TEMP,UX,..RX,…PS, PSX,…'
PFPLOT (PFP) 'Zone 1-BC,2-res,3-IC', 'DOF:TEMP,UX,…' , Size

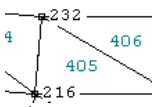
While the NFPLOT command plots only marks at nodes with a prescribed boundary condition for the specified DOF, the PFPLOT command shows *values* of nodal parameter at keypoints PT (the nearest node is found for every point PT and the corresponding value of DOF is presented on the analogue indicator of a specified *Size* (0-small,1-middle size,2-…))

EPLOT (EP) first element, last element, increment, fill, group plots elements, parameter *fill* enables to distinguish element groups (EGROUP,RCONST,MPROP) by colours and the last parameter *group* selects only elements having specified group index for plotting.
DEFPLOT plotting deformed structure
GTC number of graphs, graph 1, graph 2,… graphs of time courses selected from the matrix of observation points (it is assumed that time is in the first column of the matrix). Axis range in the window *Model* is automatically adjusted.
GTD logarithmic plot of a time curve in auxiliary window
LA write a text (label) into a graphic window
TCPLOT (TCP) index of graph, line type, colour plots another graph (time course) to the existing graphs in the window Model. No rescaling is carried out.

- The following commands pick nodes, elements,… by mouse in the model window and their characteristics are reported in the status window. *Left click* confirms selection, *Right click* terminates the commands.

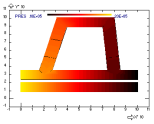
NIDENT (NID) 'Zone 1-BC,2-res,3-IC', 'DOF:TEMP,UX,…' node selection. Nodal parameters from the zone BC-boundary conditions, RES-results, or IC-initial conditions are reported.
PIDENT (PID) point (PT) selection using mouse, coordinates are reported.
CIDENT (CID) curve selection using mouse, defining points and information about the grid are reported.
EIDENT (EID) '0-general,1-Egroup,2-Rconst,3-Mprop,4-EPAR,5-geometry,6-modify' selection of element by mouse. Information concerning either parameters EGROUP, RCONST, MPROP or postprocessing results are reported in the status window.
TIDENT (TID) selection of graph (Time Course) by mouse. Basic characteristics (moments) and information on the nearest point of graph are reported.

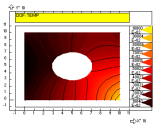
- Activation of numbering and element groups

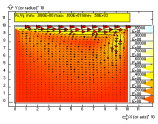


ACTNUM '1-nd,2-el,3-pt,4-cr,5-sf'
INACTNUM '1-nd,2-el,3-pt,4-cr,5-sf'
ACTSET '1-egroup,2-mprop,3-rconst','group no.' groups assigned to subsequently generated elements

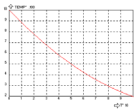
- Graphical representation of results stored in zones of node parameters

GD1 (G1)  'DOF:TEMP,UX,..RX,…PS, PSX,…','[1-BC,2-result,3-init]'
Values of selected nodal parameter are represented by varying colours (black min., bright yellow max.) inside one-dimensional elements drawn as rectangles. The second parameter specifies the nodal parameters zone: 1-BC boundary conditions (it rather serves for inspection of given values), 2-current results, 3-initial conditions.

GD2 (G2)  ,DOF:TEMP,UX,..RX,…PS, PSX,…','[1-BC,2-result,3-init]'
Contours of a specified node parameter for two-dimensional model (with triangular or quadrilateral elements). Elements are filled by colour which corresponds to the value of nodal parameter (black min. , bright yellow max.) and contour lines are plotted simultaneously.

GVEL  '[1-BC,2-result,3-init]',xmi,xma,ymi,ymax,plot mesh
Plot velocities V_x, V_y using arrows and contours of velocity.

GCR (GC) ‚Index of curve‘,‘DOF:TEMP,UX,..RX,…PS, PSX,…‘,‘[1-BC,2-result,3-init]‘

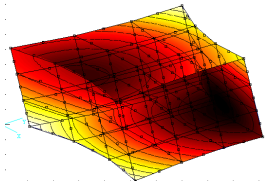


Course of a specified nodal parameter along selected curve. It is XY graph where the parameter *t* of parametrically defined functions (0 až 1) is on the horizontal axis. Nodes having distance within the tolerance TOL from the curve are used.

GFCR (GFC) ‚index of curve‘,‘index of function‘

Plots a function defined by the expression, usually using the functions *DOF*(*type of node parameter, zone number from 1 to 4, x,y,z*) or *EPA*(*type of element parameter, x,y,z*), along a specified curve.

GFSF (GD3) ‚index of surface‘,‘index of function‘,‘minimum‘,‘maximum‘



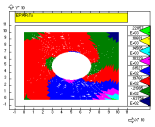
Contours of a function on a surface. The function is defined by FUNDEF and can include system functions *DOF*(*type of node parameter, zone number 1 to 4, x,y,z*), and *EPA*(*type of element parameter, x,y,z*). Unlike previous operations the contours are plotted directly in the main window MODEL. GFSF is a main graphics tool for 3D models.

- Graphical representation of postprocessing results stored in zones of element parameters

GE1 (GRAPS) ‘ENAlpha, ERE (Re), EQ (flowrate),…‘

GE1 plots results of one-dimensional problems (rotationally symmetric shells or pipeline networks). The index of EPAR matrix column is specified by name (e.g. ERE is Reynolds number, EQ-flowrate).

GE2 ‘ETX,…EII (invariant), EMISE (Mises), ESXX (stress),…‘



Same operation as GE1 but for two-dimensional problems with triangular or quadrilateral elements. Every element is filled with colour corresponding to the value in the specified column of EPAR matrix. Element parameters can be identified by name (e.g. ETX is dT/dx)

- Graphical representation of results from previous time steps stored in files *.OUT, *.TEP

GRATIM (GT) ‘Recorded parameter DOF‘ after issuing name of nodal parameter, the selection of nodes by mouse should follow similarly as in the PT operation. Graphs of time history of specified nodal parameter in selected nodes are plotted (reading file with results from all time steps *.OUT). Animation of time steps during reading the file is done in the same way as with the GD2 operation (contour lines are plotted) and after that the GRAPH window switches to X-Y display and the graphs are plotted. Note: The name of OUT-file can be changed using the command NAME problem.

LOADTN (LTN) ‘Results from time [s]‘

This operation reads file *.OUT, searches results which corresponds to the specified time (nodal parameters and integral parameters POWERE, POWERF,… defining total power of electrical field, power dissipated in flow and so on), and transfers them to the zone of boundary conditions. This command can be used for restart of computation or when we want to display results at a specific time step by using the GD2 operation.

LOADTE (LTE) ‘Results from time [s]‘

Similar operation like LTN, this time transferring results from *.TEP file, i.e. element parameters, for example gradients, flowrates,…

- Graphical representation of functions predefined by expressions or tables

GRAFUN (GF) ‘Function (-10:50)‘,from,to serves as a check of correct definitions of functions defined by tables (CURDEF) or by an expression (FUNDEF).

3.6. Setting windows

SETWIN Changes in windows configuration file $FEMWIN.CFG – size and positioning of windows, font size and type (Times, Arial, Courier,…).

SCALE xmin,xmax,ymin,ymax automatic scaling is used when zero values of x_{min},… are specified

VIEW projection, x, y, z defines projection for 3D. Default value *projection*=0 means rotation of coordinate system with respect to *x,y,z* axis, where parameters *x,y,z* are rotation angles specified in degrees. The objects, seen in the window MODEL, are in fact parallel projections to the X,Y plane of the rotated

coordinate system.The value *projection*=1 activates axonometry (axis Y is always vertical) and parameters *x,y,z* are coordinates of observer's eye.

CLS — clear screen

ZOOMI (ZI) — zoom into the plotted picture using locator

ZOOMOUT (ZO) — replots the original size (full scale)

MA — magnify (+ -) and pan

GRIDON x0,y0,dx,dy — set the grid on (entering coordinates using mouse)

GRIDOF — set the grid off

WINDOW *window number* — The standard graphic output of contours or animations are displayed in the window number 21 (VIEW). The size of this window is relatively small so the graphical outputs can be redirected to other already opened windows (e.g. MODEL or HELP).

STATUS '1-FEM,2-Model,3-time curve' — Status window is set according to type of analysis by default, but can be changed to display basic information either about Finite element model, Active model of ordinary differential equations, or about matrix of observation points.

3.7. Geometry definition

PT PT index,x,y — definition of a keypoint

There are two possibilities: either all three parameters are specified on the command line (index and coordinates) or just the index is entered from keyboard and coordinates are selected by mouse. The second method can be used to enter an arbitrary number of points with increasing indices (a new point is created with a click of left mouse button) and the sequence of points is finished by the right mouse button. The command PT for each created point is written into session file (file *.SES).

ZD Z-coordinate — z-coordinate which is automatically assigned to coordinates X,Y defined by commands PT and ND (point and node).

CR2PT (C2P) CR index,PT1,PT2 — line definition using existing points

CR3PT (C3P) CR index,PT1(left),PT2(right),PT3(mid) — curve definition using existing points

CRSPOLY (CRS) PT index,CR index — simultaneous definition of points and curves with mouse

CRNPT (CNP) CR index,PTfirst,PTlast — sequence of curves from existing points

CT contour,no.of curves,cr1,cr2,… — contour as a list of curves (can be defined by mouse)

CIRCLE (CIR) CR first,PT-center,PT-zero degree line,n-segments — Circle in X,Y plane approximated by *n*-quadratic curves

ARC CR first,PT-center,PT-zero degree line,angle,n-segments — circular arc

PTCR2 CR-first,CR-second — intersection of two curves (0,1, or 2 new points will be created)

SF3PT (S3P) SF index,PT1,PT2,PT3 — definition of surface from 3 points (triangle)

SF6PT (S6P) SF index,PT1,PT2,PT3,pt4,pt5,pt6 — definition of surface from 6 points (curved triangle)

SF4PT (S4P) SF index,PT1,PT2,PT3,PT4 — surface definition using 4 points

SF8PT (S8P) SF index,PT1,PT2,PT3,PT4,pt5,pt6,pt7,pt8 — surface using 8 points (curved surface)

SFCR (SC) SF index,CR1,CR2 — surface definition using two arbitrary curves (opposite or adjacent sides)

SFEXTR VL index,SF index,dx,dy, dz — volume by a surface extrusion

VL8PT (V8P) VL index,PT1,PT2,…,PT8 — volume using 8 points (vertices)

VLCR (VC) VL index,CR1,CR2,CR3 — volume using three arbitrary curves (edges)

VLSF (VS) VL index,SF1,SF2 — volume using two arbitrary surfaces (sides)

PTDEL,CRDEL,SFDEL,NDEL,EDEL — deletion of already defined entities (point, curve, surface, node, element)

PMERGE — merge coincident points, curves and surfaces.

3.8. Properties definition

MPROP (MP) Mprop group no., Element type, list of parameters according to the element type.

For each material parameter a pair of numbers are given: the parameter value and the number of function of time, temperature, deformation rate or coordinates, which multiplies the parameter value (function number 0 means identity – the corresponding parameter is constant)

EGROUP (EG) Egroup no., Element type, list of integer parameters according to element type

RCONST (RC) RC group no., Element type, list of real parameters according to element type

RCROSS (RCR) 'RCONST group','A-width','B-height' modify PIPE2D for rectangular cross section

Remark: At some types of special elements (heat exchangers and pumps) the meaning of EGROUP and RCONST parameters is specific and therefore specific commands RCHEX (general heat exchanger), RCHEV, RCEHEV (chevron type plate heat exchanger), RCPUMP, RCEPUM should be used instead of EGROUP and RCONST commands. Difference between e.g. RCPUMP and RCEPUM is that the first concerns any (maybe not yet existing) element but the second defines properties of a specific element identified by its index.

ERMSF SF index,EGROUP,RCONST,MPROP Change of EGROUP,RCONST,MPROP of elements inside a surface (SF).

ERMOD Sphere-Cube,radius,EGROUP,RCONST,MPROP Change of EGROUP,RCONST,MPROP of elements inside a sphere, or inside a cube (parameter RADIUS defines the sphere radius, or half-length of the cube side, respectively).

ERMEL Element,EGROUP,RCONST,MPROP Change of EGROUP,RCONST,MPROP of selected element (element is identified by its index, or using mouse optionally).

3.9. Definition of functions

FUNDEF (FDEF,FD) Index of Function (-10:50),f(TIME,TEMP,XX,YY,UX,UY,II,…)
These functions define initial conditions (arguments are usually XX,YY, or temperature TEMP as well) , boundary conditions (e.g. dependency of boundary temperature on time TIME), thermophysical properties (with respect to TEMP or the deformation rate II), and hydraulic characteristics of pipeline elements (DP-pressure loss, RE-Reynolds, DE-equivalent diameter, HE-length of element).

CURDEF (CDEF,CD,TABLE,TAB) Index of table (-10:50),'0-time,1-x,2-y,3-z,4-temp,5-ux,6-uy,7-uz,8-II',
No of points (max 8),
x1,y1,x2,y2,x3,y3,x4,y4,x5,y5,x6,y6,x7,y7,x8,y8
Note: Either an interpreted function or a table can be used for a function description. The table is restricted to only one argument, however is more suitable for complicated (e.g. discontinuous) functions.

FUNVAL (FV) variable,index of function assign variable=value of user defined function. Using a loop command you can thus tabelate a function for example in the MOP (e.g. fv c5(i) 1 for function 1).

INTGCR (ICR,IC) Index of CR, Index of function

INTGSF (ISF,IS) Index of SF, Index of function
Integral along the curve CR or the surface SF is calculated numerically, number of integration points is in the variable NINTG, result is in the variable INTEGRAL. The integrated function can be defined as an expression (FUNDEF), a table (CURDEF) or even as a program. The functions *DOF*(*type of node parameter, number of zone from 1 to 4, x,y,z*) or *EPA*(*type of element parameter, x,y,z*) can be used within the function definition as soon as the BOX command has been issued.

INTGEL (IE) Index of DOF, Index of function,first element, last element, element step
Integral over specified range of elements of DOF multiplied by a user defined function. In this case it is possible to integrate nodal parameters without necessity to execute the cartesian boxing operation BOX.

3.10. Element meshing

NMERGE coincident nodes (the nodes having relative distance less than TOL) are merged and renumbered.

Generation of nodes and elements independently of geometrical entities (PT,CR,SF,VL)

ND X, Y a new node with the next higher index will be defined similarly as the point PT (with possibility to enter coordinates with mouse).

E No of nodes, ND1, ND2, ND3,… a new element is generated on basis of already created nodes. This command is necessary when forming non-standard elements like plate heat exchanger, but other elements, like pumps, can be also defined in this way.

PPT pt-first,pt-last,increment define particles (SPHERE) identified with a range of key-points

Mesh of elements generated inside a volume, surface, contour and curve

MVL VL index, Nx, Ny, Nz, Lx/Fx(+/-), Ly/Fy(+/-), Lz/Fz(+/-), NUE(4,8,20),NIN

Elements and nodes with allocated nodal parameters are created according to the analysis type inside a specified volume. New elements (e.g. SOLID, FLOW3D) will be associated with the active groups EGROUP, MPROP and RCONST (the active group is usually the last defined group, however it can be activated explicitly by the command ACTSET). The parameter *Lx/Fx (Last/First)* affects compression of elements towards one side (positive value) or symmetrically towards both sides (negative value). NUE is number of nodes of generated elements: either 4, 8 or 20 for curved isoparametric elements. NIN is number of internal nodes of element (not used yet).

MSF SF index, Nx, Ny, Lx/Fx(+/-), Ly/Fy(+/-), NUE(3,4,5,6,7,8,9),NIN

Generation of elements (e.g. PLANE2D, FLOW2) inside a surface, according the same rules as for volumes. NUE is number of element nodes which mostly determines whether it is triangle or quadrilateral element; ambiguous situations are resolved by the sign of NUE – negative value is used only in the triangle with mid-side nodes and not without nodes at vertices (this element is used for modelling of flow). NIN is number of internal nodes, for example the node in the mass centre of element.

CRN CR,Nx,Last/First

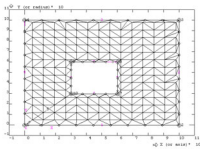
Predefines position of nodes on a curve CR (Nx-number of nodes, Last/First-uniformity). Nodes are not genarated but this information is used as default in MVL,MSF,MCR commands and first of all in the automatic meshing by using MCT and MRG.

MCT no.of contours, Nx, Ny, NUE(3,4,6),NIN

Generation of triangular elements (e.g. PLANE2D, FLOW2) inside a contour. This method is recommended only for simple regions without internal holes (no.of contours=1).

MRG index of contour, NUE(3,4,6), NIN MRG, Nx, Iterat, N-division, A-bound,C1,C2

Generation of triangular elements (e.g. PLANE2D, FLOW2) inside a region. This method is suitable for multiply connected regions with internal holes, defined as a part of a single contour:



MPT PT-first,PT-last,PT-step,NUE(3,4,6),NIN

Generation of triangular elements connecting a set of key-points PT.

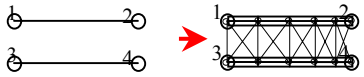
MND ND-first, ND-last, ND-step,NUE(3,4,6),NIN

Generation of triangular elements connecting a set of existing nodal points PT.

MCR CR first, Nx, Last/First(+/-), NUE(2,3), RC-group, CR last, CR step

Generation of elements on curves *CR-first* up to *CR-last* with step *CR-step*. The parameter *RC-group* associates new elements with the specified RCONST group, no matter which one was active (default value *RC-group*=0 associates obviously elements with the active group). Elements generated using this command are for example PIPE2D, TRUSS2D, BEAM2D, HEXC, VALVE, PUMP, CSTR, SHELLAX as specified by the appropriate EGROUP parameter.

MCR2 CR pipe 1, CR pipe 2, Nx, MPROP-pipe 1, MPROP-pipe 2, RC-pipe 1, RC-pipe 2, RC-HEXC



Nx pairs of PIPE2D two-node elements of the same length are generated on curves 1 and 2 (groups of real constants and material properties are given explicitly) and simultaneously *Nx* of four-node elements HEXC (heat exchanger) which connects corresponding PIPE2D elements are created (the group of real constants is given explicitly to specify heat transfer area among others). Indices of PIPE2D elements creating the exchanger are stored in the zone of element parameters IEPAR for each 4-node elements (the last two columns of IEPAR matrix are used) – this arrangement speeds up computations.

MCR4 CR pipe 1, CR pipe 2, RC-HEXC

creates four node elements HEXC, the same as elements generated by the operation MCR2. The only difference is that only the 4-node elements, which represent a connection of two-node already existing PIPE2D elements, will be generated (the elements PIPE2D had to be generated by MCR before).

Structure of parameters in the EGROUP and first all in the RCONST group for elements HEXC and PUMP is different than for other types of elements. Therefore special commands for RCONST definition of heat exchangers are available

RCHEX for tubular heat exchangers created by the command MCR4.
RCHEV for chevron type plate heat exchangers defined by command E (4 node element)
RCEHEV the same as RCHEV however indices of groups RCONST and EGROUP are not specified explicitly, but via the index of element, which has to be created before (the command is intended rather for change of yet defined parameters of a heat exchanger)

When selecting a centrifugal pump from database (elements PUMP) it is necessary to specify some estimated operating parameters (flowrate, displacement height, dimensions of nozzles) and kind of application (feeding pump, slurry pump,..) using commands

RCPUMP (RCP) 'Pump 0-any,1-milk,2-waterworks,...','Nozzle diameter [m]','Estimated flowrate [m^3/s]', 'Estimated discharge height [m]'
RCEPUM (RCEP) modification of parameters for existing element PUMP (usually selection of another pump from database)

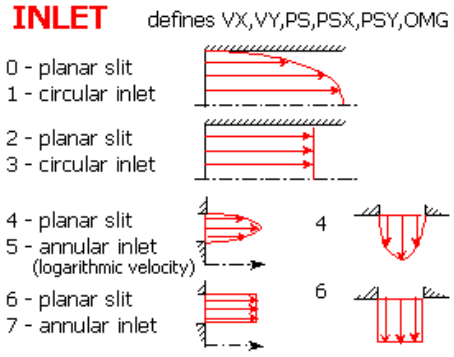
3.11. Boundary conditions and loading

Specifying values and attributes of selected node parameter in nodes on a curve or on a surface (inside the surface, not on its boundary). This way strong boundary conditions (nodal parameter status -10,...,-1) and load (status 1,...,10) or weak boundary conditions of the 3rd kind (status>20) are defined.

NFCR (FCR) ,CR index‘, ,DOF:TEMP,UX,..RX,…PS, PSX,…‘,‘status (BC)‘, ,P1(left)‘,‘P2(right)‘,‘P3(midpoint)‘
Example of the velocity profile definition FUNDEF –2,1.5*(1-YY**2) (function no.-2, e.g.parabola)
NFCR 17,VX,-2,1,1,1 (velocity u_y on curve 17)
Example of heat transfer boundary condition TE(1)=20 (ambient temperature)
NFCR 10,TEMP,21,200,200,200 (α=200 on curve 10)
NFSF (FSF) ,SF index‘, ,DOF:TEMP,UX,..RX,…PS, PSX,…‘ ,‘status (BC)‘, ,P1‘,‘P2‘,‘P3‘,‘P4‘,‘P5m‘,‘P6m‘, ,P7m‘,‘P8m‘
NF (F) ,ND index‘, ,DOF:TEMP,UX,..RX,…PS, PSX,…‘ ,‘status (BC)‘,‘Value‘
NFPT (FPT) ,PT index‘, ,DOF:TEMP,UX,..RX,…PS, PSX,…‘,‘status (BC)‘,‘Value‘

For fluid flow it is possible to simplify specification of velocities, stream function, vorticity by using commands

INLET ,CR index‘, ‘status (BC)‘, ‘u-mean velocity’,’profile’
WALL ,CR index‘, ‘status (BC)‘



The commands INLET and WALL calculate not only velocities but also stream function PS and its derivatives. Be careful about the order of these command if you are going to use the stream function. You should start with INLET at axis (at axis the value PS=0) and continue along the boundary – newly defined section always starts with the stream function value calculated in the previous step (this value can be checked or modified because it is system variable PSWALL).

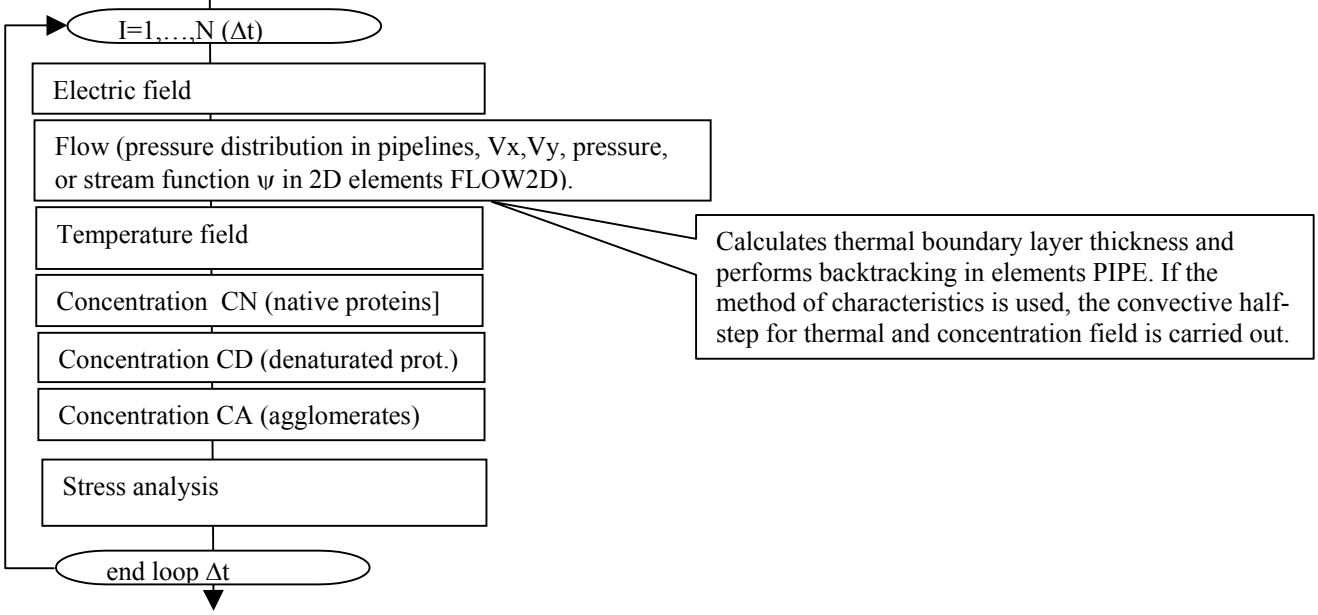
Initial conditions assignment (they are zero by default)

INITIA (INI) ,DOF:TEMP,UX,..RX,…PS, PSX,…‘, ,Function (-10:50)‘
Note: When zeroing initial conditions it is sufficient to specify function number 0 (even if it is not defined)

3.12. FEM solution

Solution of FEM problems is executed by the independent program RUNFEM.EXE: First the reduced FEMINA database is written to a working binary file, then the control is transferred to the external program RUNFEM.EXE (the program modifies database and creates output files with results and information about processing) and finally the modified database is read back to FEMINA for postprocessing. This sequence is repeated many times automatically if a problem of optimisation is solved using OPTIMA or SOMA algorithms implemented in FEMINA.EXE.

RUNFEM solves unsteady-state problems of fluid flow with simultaneous heat and mass transfer, chemical reactions and volumetric source of heat. Solver operates with 4 zones, it means that four real numbers are reserved for each DOF (nodal parameter, e.g. temperature): the first zone are values specified by user (usually boundary conditions and loads), the second is a working zone for frontal solver (temporary results), in the third zone are initial conditions (updated in each time step) and results of previous iteration are in the last zone. A sequence of partial problems is solved at every time step and at every time level iterations can proceed until a prescribed accuracy is achieved. Specification of zero iteration number excludes the relevant operation from the whole sequence.



After the iterations terminate the postprocessing begins giving as a result element parameters (EQ[4]-flowrate, ETAU-shear stress, ERE-Reynolds number, EFOUL-thermal resistance of fouling layer, EBL- thickness of thermal boundary layer, EDISS-dissipated power, ETX-temperature gradients, ENA-normal forces, EMA-bending moments, and so on). At the end of time step, calculated values of nodal and element parameters are written into OUT and TEP file respectively, together with calculated integral quantities POWERE, POWERF,…,POWERS (total electric power, heat dissipated by friction, …, strain energy).

All these activities are initiated by the command SOLVE[5]

SOLVE ,Initial Time‘,‘No. of time steps‘,‘dt [s]‘, ‘Append EPA‘,‘iter electric‘,‘iter flow‘,‘iter thermal‘, ‘iter concentration CN‘,‘iter concentration CD‘,‘iter concentration CA‘, ‘iter static‘

Most parameters are self-explaining, only the meaning of parameter “Append EPA” is little bit more complicated: Default value 0 is used when new solution is to be started, because element parameter zone is cleared (therefore flowrates, gradients, fouling layer thickness and so on are set to zero) and new empty output files will be opened. Value 1 activates the same actions, only the element parameter zone is not initialised and therefore results from the previous solution step (e.g. calculated flowrates) can be used for solution. Value 2 skips initialisation completely and solution is a continuation of the previous one, with results appended to the old result files (OUT and TEP).

[4] The names written with capitals, are names of system variables, see later.

[5] The only exception is stress analysis of structure composed from trusses by the Monte Carlo method. In this case the command SOLVE is to be replaced by the command TRUSS.

DSOLVE	the same as SOLVE, but using double precision solver RUNFEMD.EXE.
PSOLVE	particles (SPHERE,CYLINDER) transferred in flow-field.

Different methods can be selected for the solution of flow, heat and mass transfer. The methods and other options can be specified by using the command OPTION or in the dialog window *Option* (see 2.2.1) from menu *Solution*, where most menu items are common for 1D, 2D and 3D problems, nominally buoyancy (natural convection using Boussinesque approximation or simply by definition of temperature dependent density), upwind (selection Galerkin/Galerkin Petrov), desired accuracy, number of iterations, underrelaxation, implicit/explicit time integration, and frontal method settings. However some options are specific for 1D and 2D problems: For 1D problems with pipes it is possible to select (see parameter *Solution Method* in *Options*) either weighted residual method (default 0), enthalpy balances (1) or method of characteristics (2), it is possible to select a fouling model and corrections of thermal boundary layer (parameter BL-Boundary Layer). On the other hand for the 2D transport equations the option *Solution Method* represents only the choice between the weighted residuals and method of characteristics, while enthalpy balances, as well as boundary layer and fouling are irrelevant.

Some options are postponed to the element level as integer EGROUP parameters. Only few of them are common for all types of elements: the name of element, decision between steady or transient solution (some elements can be considered in equilibrium, others not) and volumetric heat sources defined by a function. However the meaning of most EGROUP parameters differs in 1D and 2D elements: For example the parameter METHOD selects a pump in the 1D element PUMP, defines a type of a heat exchanger in the 1D elements HEXC, and the same parameter METHOD selects one of the following methods used for the solution of fluid flows in the 2D elements FLOW2D:

CREE (CREEP)	creep flow – stream function ψ $\psi_{,x}$ $\psi_{,y}$ (nodal parameters of the type PSI, element T3)
PSIN	Navier Stokes, stream function ψ $\psi_{,x}$ $\psi_{,y}$ (nodal parameters PSI, element T3)
CARE[6]	Navier Stokes, stream function ψ, vorticity ω (nodal parameters PSI, elements T3,T6,Q4,Q8)
UVP	Navier Stokes, velocities u_x u_y pressure p, incompressible liquid (nodal parameters UVP, element T6)
UVPP	Navier Stokes, velocities u_x u_y pressure p, pseudo-compressibility (nodal parameters UVP, element T6)
PENS[7]	Navier Stokes, velocities u_x u_y , pressure elimination by the penalty function (nodal parameters UVP, elements T3,T6,Q4,Q8)
MIKE	velocity field u_x u_y with minimal kinetic energy, pressure p is determined by the Lagrange multipliers method. Nodal parameters PSI or UVP, elements T3,T6,Q4,Q8.
MIDE	velocity field u_x u_y with minimal dissipation and kinetic energy, and with minimal squares of the continuity equation residue (least square method). Nodal parameters PSI/UVP, elements T3,T6,Q4,Q8.

There exists a special finite element USER with arbitrary number of nodes (e.g. 1 for mass, 2 for a pipe, 3 for a triangle, and so on) which can be defined by user. It means that there exists the Fortran subroutine USER(ICNTR,IOPER,IE,NL,NUE,AL,BL) in the file $USER.FOR which is to be completed, translated and linked to the executable program RUNFEM.EXE. The parameters ICNTR,IOPER,IE,NL,NUE are inputs, and arrays AL(NL,NL), BL(NL) are outputs, local matrix and right hand side vector of element with the name USER. The most important input parameter is IOPER(=1 electric field,=2 flow field,=3 temperature field,=4,5,6 concentrations,=7 static analysis), IE-index of element and NUE-number of nodes. It is assumed that there is only one unknown parameter in each node (voltage, pressure, temperature or concentration, depending upon IOPER) with the exception of static analysis, and therefore dimension of element matrix NL equals NUE. When defining arrays AL(NL,NL) and BL(NL) the following subroutines make an access to the FEMINA database easier

SUBROUTINE GETXYZ(IE,X,Y,Z)	coordinates X(*),Y(*),Z(*) of nodes of element IE
SUBROUTINE GETDOF(IE,IDOF,DOFV)	nodal parameters DOF(*) of element IE
SUBROUTINE GETEPAR(IE,IST,IGAUS,IMETH,H,D,RKX,RCP,RHO,RKAPPA,VISC,RE,QFLOW,VX,VY)	

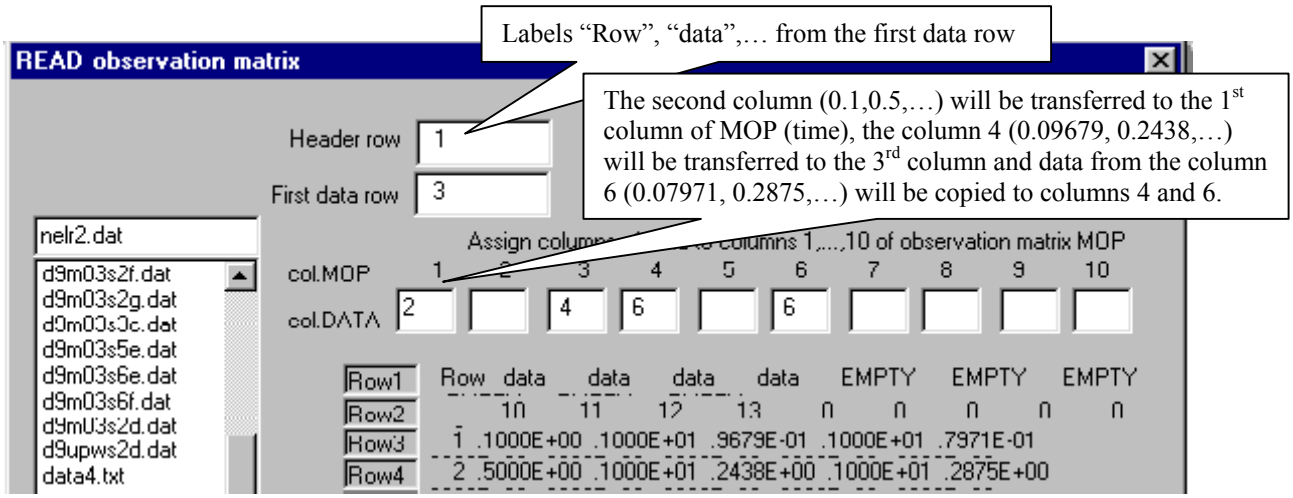
[6] CARE – according to name of the method inventor, madame CAmpion-REnson
[7] PENS – PEnalty Navier Stokes

3.13. Time series processing

Time curves are results of transient solutions or experiments, and several operations are available for their processing: evaluations of characteristics (moments) of time courses, smoothing, interactive editing, comparison, etc. Time courses are stored in FEMINA database as a table t_i, y_i called matrix of observation points MOP. The first column of the table (time) constitutes a common time base for all function profiles. Maximum number of points is 5096 and maximum number of columns is 10 – individual time courses are labelled by indices from 2 to 10.

The following operations enable data import, editing and transformations of MOP:

REOMAT (ROM) reads selected columns of MOP from a file with postfix TXT or DAT. The data file is selected from menu and the selection is enhanced by continuous displaying the content of browsed files. After a file is selected it is necessary to specify the row containing column labels (used for headers) and the rows with data which are to be transported into MOP, see the following dialog window:



TSTEP 'No. of time steps','Time step dt [s]' this command defines in fact a new MOP

TCEDIT (TCE) editing points of a selected graph with mouse (several regimes of edit are available)

TCINPUT (TCI) entering items into selected column of MOP directly from keyboard

TCSET (TCS) specification of column types and headers

TCLIST (TCL) lists the observation matrix (10 columns in the *Model* window)

Column definition of MOP by an algebraic expression

TCF assignment Variables C1,C2,…,C10 are reserved for column referencing and together with the index I of an observation point (row index of the observation matrix) can be used as part of the command. Examples: C1=0.1*I defines equidistant time base (times in the first column will be 0.1, 0.2, 0.3,…). This command can be used also for a half life decay correction of a response and for example TCF C2=C2*EXP(0.0005*C1) corrects response in the second column.

TCFR row1,row2,assignment

Same as TCF but the assignment is applied only to the specified range of rows.

Comparison of selected pairs of columns

COMPAR Number of TC pairs, Index of first TC in pair 1, Index of second TC in pair 1,…[8]

Specifies pairs of responses (prediction – experiment) and weights which are used for calculation of deviations between the model prediction and measured responses profiles.

CRITER Criterion for TC comparison

[8] TC is to be read as Time Course.

Choice of a deviation norm (this norm is used when deviation of several response pairs is calculated, see COMPAR and TCDEV).

0: $\sum_{i=1}^{N} |y_1(t_i) - y_2(t_i)| / N$, 1: $\sum_{i=1}^{N} |(y_1(t_i) - y_2(t_i)) / \max(y_1(t_i), y_2(t_i))| / N$

2: $\sqrt{\sum_{i=1}^{N} (y_1(t_i) - y_2(t_i))^2 / N}$, 3: $\sqrt{\sum_{i=1}^{N} ((y_1(t_i) - y_2(t_i)) / \max(y_1(t_i) - y_2(t_i)))^2 / N}$

4: $(\int_0^T |y_1(t) - y_2(t)| dt) / T$, 5: $(\int_0^T |(y_1(t) - y_2(t)) / \max(y_1(t), y_2(t))| dt) / T$

6: $\sqrt{(\int_0^T (y_1(t) - y_2(t))^2 dt) / T}$

TCDEV calculates and reports deviation for actual settings (COMPAR and CRITER); result is system variable SCOMP.

COPYTO (CTO) Copy TC, to TC nearly the same as e.g. TCF C2=C5, but labels of graphs are copied too.

Integral characteristics and normalisation

MOMENT Index of TC Calculation area, first and second moment (gravity centre and variance)

FMOM Index of TC, index of point (tail) calculates moments with tail shifting (see FFT analysis)

NORM Index of source TC, Index of normalised TC, '0-unit area,1-unit mean time'
Normalisation of a time course to the unit area or even to the unit first moment.

TCINTG Index of source TC, Index of integrated TC
Integration of time course (e.g. calculation of integral residence times distribution F)

Predefined impulse responses

IDMSER Index of new TC, No.of mixers (real!), Mean residence time
Generation of the impulse response of N-ideal mixers with specified mean residence time. The parameter specifying the number of mixers need not be integer number.

PASERI Index of new TC, N1, N2, f=Q1/Q, alfa=T2/T1, Mean residence time
Generation of the impulse response of two parallel series N1 and N2-ideal mixers with specified mean residence time of the whole system. The parameter *f* is a relative flow rate and the parameter α is a ratio of residence times in parallel branches.

Time curve processing

SMOOTH Source TC, Smoothed TC, '*Method* 0-linear,>0-quadratic,<0-median'
Smoothing a time course based on local substitution of a function profile by linear or quadratic function. The coefficients of approximation polynomials are determined on the basis of the least sum of squares of deviations in specified number of points (this is the parameter *Method*, number of points before and after the current point) and also from the continuity requirement in connecting points. Instead of polynomial smoothing the median smoothing of 2N+1 points is used if the specified parameter *Method* is negative (N-points before and N-points behind the i-th point of the table is used for median calculation). The time range of smoothing is determined by mouse.

TCRND Source TC, NOISED TC, 'Noise 0-absolute,1-relative', 'mean amplitude'
Generation of artificial white noise using a pseudo-random generator with Gaussian distribution (parameter „mean amplitude“ is a variance of Gaussian distribution).

TAIL Source TC, Corrected TC, 'Tail: 0-exp(t), 1-t.exp(t), 2-A/t^3'
This operation substitutes the tail of selected graph by exponential or hyperbolic functions (which corresponds to diffusive and convective flows, respectively). The range of points, which are used in the regression analysis to identify the tail, is selected with mouse as well as the point from which the replacement should be applied. Because the approximation function is Constant + asymptotic function fading away, the result of the regression analysis is also this constant and so the TAIL operation can be used for finding of a vertical shift of the response function which corresponds to a background. Thus the tail approximation can be used also for the background correction.

TCR Source TC, Corrected TC, model Regression a part of TC (specified by mouse) by a model

TCSUB Source TC, result TC, subtracted TC subtraction of a time curve

TCJOIN Source TC, result TC, inserted TC patch a time curve by a part of TC identified by mouse

TCLIP Index of graph cut off negative values of the response (they will be replaced by zero)

TCBGR Source TC (Time Course index), corrected TC (index)
Correction of time course compensating gradually changing (raising) background during experiment.

TCYSHF(CS) Source TC, Modified TC, Shift vertically by an increment
Shift a time course in the vertical direction by a specified (positive or negative) value. If the specified value is zero (default), mouse is activated.

TCXSHF(TS) Source TC, Modified TC, Shift horizontally by an increment
Time shift by a specified (positive or negative) value.

TCOSHF(TCS) Source TC, Modified TC, Horizontal shift, Vertical shift
Shift a time course to a new origin (shift is done in horizontal and vertical direction).

TCEND(TCRANGE) t_{min},t_{max}
Set a new common time range for all time curves by mouse.

TCZERO(TCZ) Source TC, Modified TC
Fill zeroes into specified part of a time course.

TCDT(TCT) n_{steps} , Δt
Recalculate all time curves to a constant time step.

TCFFT (FFT) 'Deconvolution E=Y/X (1), convolution (2), cross-correlation (3), filter (4)', 'Column X', 'Column Y', 'Col. E','Smoothing'
Fast Fourier Transform applied to data in MOP (max.5096 points) calculates Fourier coefficients and power spectral density (PSD) of resulting function. Using PSD graph it is possible to select threshold frequency and suppress high frequency noise. More details are explained in chapter 4.

3.14. RTD models

RUNMOD program is focused on flow systems described by systems of ordinary differential equations (sometimes called as compartment or "lumped parameter" models). This group of operations is not based on the finite element method but a numeric integration using Runge-Kutta methods.

RMODEL 'Model name'
Reads model from the text file which describes a set of ordinary differential equations. The model is defined by a simple script language with the same syntax as the language used for commands and parameter interpretation by FEMINA program (even same predefined system variables can be used).

Example:

```
Ideal mixers series
IP1-number of mixers
RP2-mean residence time
\\ini
real tm
neq=IP(1)
tm=RP(2)/neq
cm(1)=1/tm
\\mod
dcm(1)=(xv(1)-cm(1))/tm
for i=2,neq do
[ dcm(i)=(cm(i-1)-cm(i))/tm ]
yv(1)=cm(neq)
\\par
idenum=1000001 method=1 inp=1 out=1 npar=2 NEQ=5
x1=1 y1=2
No.of mixers: IP(1)  default=5 min=1 max=100 relFAKT=0.1
Mean residence time: RP(2)  default=1 min=0.01 max=50 relFAKT=0.2
```

This is just an introductory text describing the model (in this case, it is an ideal mixers series with two parameters, number of mixers and total mean residence time)

Initial section of program which is executed only once before beginning of time integration. The declarations in this section are valid also for the following section (e.g. variable TM-time constant of mixer). The initial conditions are primarily defined in the INI section (e.g. initial concentration in the first mixer).

The model itself is defined in the MOD section: it describes the first derivatives of calculated variables (here, for example, concentrations CM(1),CM(2),…,CM(NEQ) with their derivatives labeled as DCM(1),…,DCM(NEQ)).

Definition of parameters (default values).
IDENUM determines graphical representation of the model, METHOD selects either Euler or Runge Kutta method for solving the set of equations, INP,OUT are numbers of inlets/outlets, NPAR-number of model parameters, NEQ-num. of equations, X1,Y1-indices of inlet and outlet functions (time courses in MOP). Then, one row of data is reserved for each model parameter.

The line begins with a text finished by colon which is displayed when the parameter is specified in dialog. Then, the parameter name follows – the vector RP is reserved for these models, however any database element referenced by the name of system variable can be used (for example, the coordinate XPT(20) of relative point, variables A,B,… or any other user defined variables can be used for the model definition). The following items are identified by keywords so theirs order is not important and they can be omitted.

Syntax of the script language used for the model definition is simple.

Some variables and vectors are predefined (called system variables, the most important are vectors CM(i),DCM(i)-values and first derivatives of concentrations, TIME, RP(i),IP(i) – model parameters, XV(i), YV(i)-current values of stimulus and response functions, respectively), but you can define your own simple variables and vectors (real or integer) using commands

REAL *list*, INT *list*, where *list* consists of identifiers (max.8 characters) separated by commas.

Assignment has a usual form

variable=expression or *variable*:=*expression*

and in the *expression* variables (like A,VV(i)), constants (like 123, -0.234e-5), parentheses ((,)), arithmetic operators (+,-,*,/,**), relational operators (=,<,>,<=,>=,<>), logical operators (&,|), and functions, see bellow, can be used with common precedence and priorities.
sin(x), cos(x), atn(x) – goniometric functions and arctg
abs(x) - absolute value, exp(x) – exponental function,
log(x) - natural, lgt(x) – decadic logarithms,
min(x1,x2,…), max(x1,x2,…), minimum and maximum (variable number of parameters)
erf(x) - error function,
ple(n,x) -Legendre polynomial, plg(n,x)-Laguerre polynomial, plt(n,x)-Chebyshev polynomial,
jn(n,x), yn(n,x) – Bessels functions of the first and the second kind of the order n,
rnd(x)-random number generator,
cvt(i,t)-interpolated value in the *i*-th column of MOP in time *t*,
xvt(i,t), yvt(i,t)-interpolated values of i-th stimulus and response functions respectively, in time *t*),

The following commands are available for the program control:

WHILE *relation* DO *block*
IF *relation* THEN *block*
IF *relation* THEN *block* ELSE *block*
FOR *i=imin,imax* DO *block*

where *block* is either assignment, DISP command, or several commands enclosed by *begin*, *end* or by brackets [,][9]. The brackets {, } enclose comments.

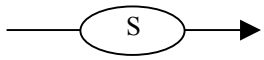
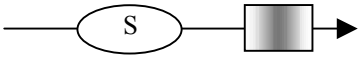
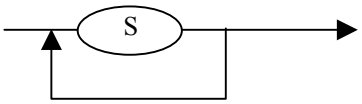
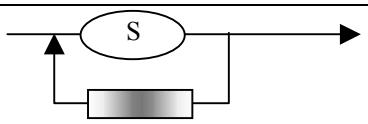
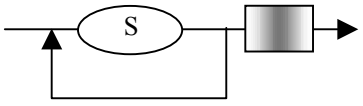
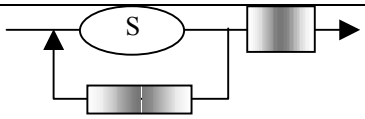
Remark: Keywords and basic entities, like assignment, relations, are separated by one or more spaces, therefore it is not possible to use spaces for example inside an expression!

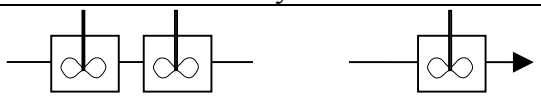
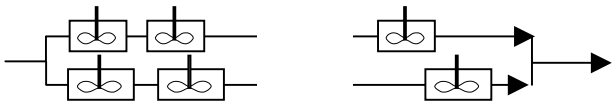
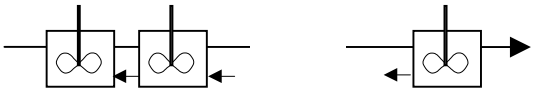
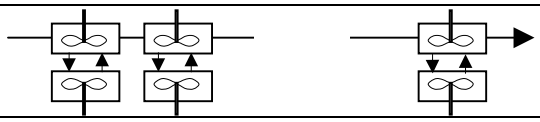
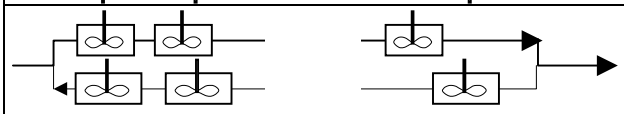
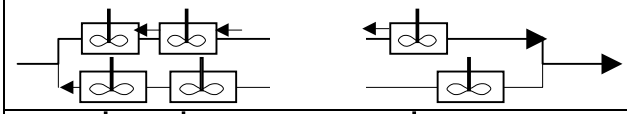
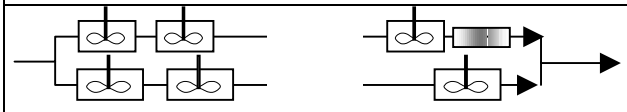
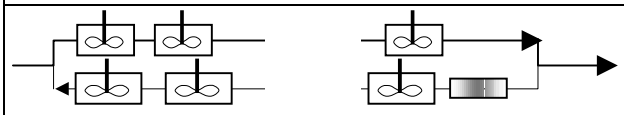
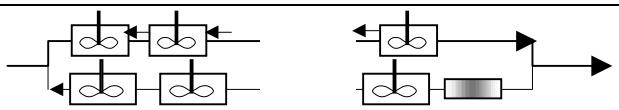
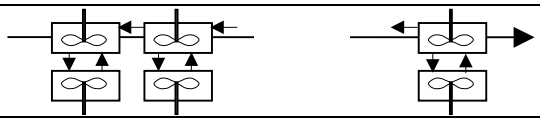
[9] *Block* is not for example the construct FOR … or the IF … clause. Therefore it is necessary to use brackets [ ] when nested loops are to be used: FOR I=1,10 DO [FOR J=1,I DO DISP I,J], or IF I<J THEN [FOR K=1,I DO DISP I,J,K].

It is not necessary to add the part *par* into the model definition file, this information can be added later for example by filling a questionnaire in the dialog window as a part of the RMODEL command processing. It could, but need not be possible also to assign a graphic scheme to a model by specifying the parameter IDENUM. In the following table we shall try to explain the way how the standard models are designed, naming conventions and meaning of individual digits in IDENUM

IDENUM: *gsrrnnmm*
mm – index of model parameter describing number of mixers in the first section
nn – index of model parameter describing number of mixers in the second section
rr – index of model parameter which specifies relative volume of the first branch (graphic representation of a model is not a static icon, it should represent actual proportions of units)
s – number identifying the base subsystem
g – number identifying „outer“ system (additional connection of piston flow or recycle)

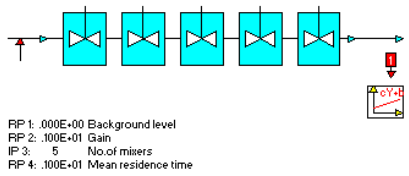
File name of model description, maximum is 4 characters: *SGC*.mdt
C – specifies a correction of detector gain and background raise (0-no correction, A-gain, B-background, L-combination of both)
G – identification of „outer“ system
S – identification of base system (1 or 2 characters)

Scheme of outer system	g	G
	0	0
	1	P
	2	R
	3	U
	4	V
	5	W

Scheme of basic subsystems	s	S
	0	S
	1	SS
	2	B
	3	D
	4	SR
	5	BR
	6	SP
	7	RP
	8	BP
	9	BD

–

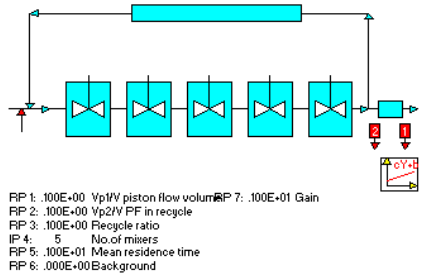
IDEAL MIXERS SERIES S0L.MDT



Ideal mixers series S0L.mdt
PM1-a1 background level
PM2-a2 scale (detector gain)
PM3-number of mixers
PM4-mean residence time

```
\\ini
real tm
neq=ip(3)
tm=rp(4)/neq
cm(1)=1/tm
\\mod
dcm(1)=(xv(1)-cm(1))/tm
for i=2,neq do dcm(i)=(cm(i-1)-cm(i))/tm
yv(1)=rp(1)+rp(2)*cm(neq)
\\par
idenum=0000003 method=1 inp=1 out=1 npar=4 NEQ=5
x1=2 y1=3
Background level: rp(1) default=0 min=-1000 max=1000 relfakt=1
Gain: rp(2) default=1 min=0.001 max=1000 relfakt=1
No.of mixers: ip(3)  default=5 min=1 max=100 relFAKT=1
Mean residence time: rp(4)  default=1 min=0.01 max=50 relFAKT=1
```

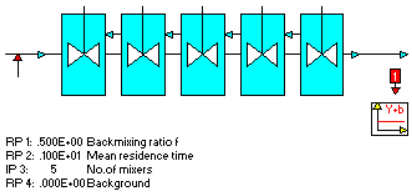
IDEAL MIXERS SERIES+PF+RECYCLE PF SWL.MDT



Ideal mixers series+PF+recycle PF SWL.mdt
RP1- Vp1/V piston flow volume
RP2- Vp2/V piston flow volume
RP3- recirculation
IP4- number of mixers
RP5- mean residence time
RP7- background
RP8- gain

```
\\ini
real tm,pf1,pf2,r,tmean,del1,del2
int i
pf1=rp(1) pf2=rp(2) r=rp(3) tmean=rp(5) neq=ip(4)
del1=pf1*tmean del2=pf2*tmean
tm=tmean/neq*(1-pf1-pf2)
cm(1)=1/tm
\\mod
dcm(1)=(xv(1)+r*yvt(2,time-del2)-(1+r)*cm(1))/tm
for i=2,neq do dcm(i)=(1+r)*(cm(i-1)-cm(i))/tm
yv(2)=cm(neq)
yv(1)=yvt(2,time-del1)*rp(7)+rp(6)
\\par
idenum=50000004 method=-2 inp=1 out=2 npar=7 NEQ=5
x1=2 y1=3 y2=4
Vp1/V piston flow volume: rp(1) default=0.1 min=0 max=100
relfakt=1
Vp2/V PF in recycle: rp(2) default=0.1 min=0 max=100 relfakt=1
Recycle ratio: rp(3) default=0.1 min=0 max=100 relfakt=1
No.of mixers: ip(4) default=5 min=1 max=100 relFAKT=0.1
Mean residence time: rp(5) default=1 min=0.01 max=50 relFAKT=1
Background: rp(6) default=0 min=0.0 max=50 relFAKT=1
Gain: rp(7) default=1 min=0.1 max=50 relFAKT=1
```

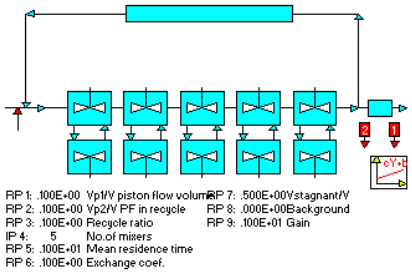
IDEAL MIXERS SERIES WITH BACKMIXING B0B.MDT



Ideal mixers series with backmixing B0B.mdt
RP1-backmixing ratio
RP2-mean residence time
IP3-number of mixers (integer)
RP4-background

```
\\ini
real tm
neq=ip(3)  f=rp(1)
tm=rp(2)/neq
cm(1)=1/tm
\\mod
dcm(1)=(xv(1)+f*cm(2)-(1+f)*cm(1))/tm
for i=2,neq-1 do dcm(i)=((1+f)*cm(i-1)+f*cm(i+1)-(1+2*f)*cm(i))/tm
dcm(neq)=((1+f)*(cm(neq-1)-cm(neq)))/tm
yv(1)=cm(neq)+rp(4)
\\par
idenum=2000003 method=1 inp=1 out=1 npar=4 NEQ=5
x1=2 y1=3
Backmixing ratio f: rp(1) default=0.5 min=0.001 max=10 relafakt=1
Mean residence time: rp(2) default=1 min=0.01 max=50 relFAKT=1
No.of mixers: ip(3) default=5 min=1 max=100 relFAKT=1
Background: rp(4) default=0 min=0.0 max=50 relFAKT=1
```

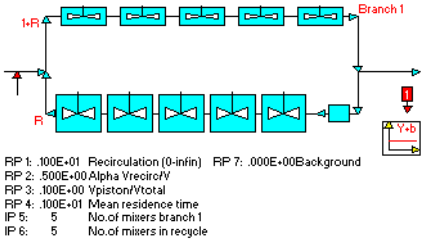
STAGNANT REGIONS+PF+RECYCLE PF DWL.MDT



Stagnant regions+PF+recycle PF DWL.mdt
RP1- Vp1/V piston flow volume
RP2- Vp2/V piston flow volume
RP3- recirculation
IP4- number of mixers
RP5- mean residence time
RP6- q exchange coef.
RP7- alfa Vstagnant/Vmixed
RP8- background
RP9- gain

```
\\ini
real tm,pf1,pf2,r,tmean,del1,del2,v1,v2,alf,q,y2
int m,i
pf1=rp(1) pf2=rp(2) r=rp(3) tmean=rp(5) q=rp(6) alf=rp(7)
neq=ip(4)*2 m=ip(4)
del1=pf1*tmean del2=pf2*tmean
tm=tmean/neq*(1-pf1-pf2) v1=tm/(a+alf) v2=tm*alf/(1+alf)
cm(1)=1/v1
\\mod
y2=yvt(2,time-del2)
dcm(1)=(xv(1)+r*y2+q*cm(m+1)-(1+r+q)*cm(1))/v1
for i=2,neq do
[dcm(i)=((1+r)*cm(i-1)+q*cm(m+i)-(1+r+q)*cm(i))/v1
 dcm(m+i)=q*(cm(i)-cm(m+i))/v2]
yv(2)=cm(m)
yv(1)=yvt(2,time-del1)*rp(9)+rp(8)
\\par
idenum=53070004 method=-2 inp=1 out=2 npar=9 NEQ=10
x1=2 y1=3 y2=4
Vp1/V piston flow vol.: rp(1) default=0.1 min=0 max=100 relfakt=1
Vp2/V PF in recycle: rp(2) default=0.1 min=0 max=100 relfakt=1
Recycle ratio: rp(3) default=0.1 min=0 max=100 relfakt=1
No.of mixers: ip(4)  default=5 min=1 max=100 relFAKT=0.1
Mean residence time: rp(5)  default=1 min=0.01 max=50 relFAKT=1
Exchange coef.: rp(6)  default=.1 min=0.01 max=50 relFAKT=1
Vstagnant/V: rp(7)  default=.5 min=0.01 max=.9 relFAKT=1
Background: rp(8)  default=0 min=0.0 max=50 relFAKT=1
Gain: rp(9)  default=1 min=0.1 max=50 relFAKT=1
```

TWO SERIES IN RECYCLE (PF IN BRANCH 2) RP0B.MI




```
Two series in recycle (PF in
branch 2) RP0B.MDT
 [RP1]- r
 [RP2]- ŕ= V2.N/V
 [RP3]- V3/V
 [RP4]- V/Q mean residence time
 [IP5]- No. of units in 1 serie
 [IP6]-              in 2 serie
 [RP7]- background
```

```
\\INIT
real alfa,beta,tm,tp,t1,t2,ydel,aux,bcgr
integer i,m,n
r=rp(1) alfa=rp(2) beta=rp(3) tm=rp(4) m=ip(5) n=ip(6) bcgr=rp(7)
neq=n+m
t1=tm*(1-alfa-beta)/m t2=tm*alfa/n tp=beta*tm/r
cm(1)=1/t1
\\MODEL
ydel=yvt(1,time-tp)-bcgr
dcm(1)=(xv(1)+r*cm(neq)-cm(1)*(1+r))/t1
for i=2,m do dcm(i)=(cm(i-1)-cm(i))*(1+r)/t1
  dcm(m+1)=(ydel-cm(m+1))*r/t2
for i=m+2,m+n do dcm(i)=(cm(i-1)-cm(i))*r/t2
yv(1)=cm(m)+bcgr
\\PARAM
idenum=07020605 method=1 inp=1 out=1 npar=7 NEQ=5
x1=2 y1=3
Recirculation (0-infin): rp(1) default=1 min=0 max=1e10 relfakt=1
Alpha Vrecirc/V: rp(2) default=0.5 min=0.001 max=10 relfakt=1
Vpiston/Vtotal: rp(3) default=0.1 min=0 max=1 relfakt=1
Mean residence time: rp(4)  default=1 min=0.01 max=50 relFAKT=1
No.of mixers branch 1: ip(5)  default=5 min=1 max=100 relFAKT=1
No.of mixers in recycle: ip(6)  default=5 min=1 max=100 relFAKT=1
Background: rp(7) default=0 min=0. max=1e10 relfakt=1
```

The parameters in the model definition file can and usually are modified by the following commands

PARDEF No.of parameters, p_1-name, p_2-name,…
model parameters selection (any system variable can be used)

PARLIM min p_1, max p_1, min p_2, max p_2,… parameter limits

PARSET p_1, p_2,….set actual values of parameters.

PARFIT p_1, *'0-supressed,1-linear,2-nonlin,3-search'*, ω_{p1}, p_2, *'0-supressed, …'*, ω_{p2},…
Determines which model parameters are to be optimisied – comparison of model prediction with experiment or with numerically calculated data (time courses). Parameters marked as LINEAR indicates a linear or approximately linear model prediction – this is used for better convergence of regression analysis. The NONLIN label indicates a nonlinear parameter and usage of the Marquardt-Levenberg method. The SEARCH specifies a one-dimensional parameter search (one-dimensional minimization of deviations between the model and data based upon golden section method).

METHOD 'M: 0-Euler,>0-RK fix.dt,<0-variab.step', 'no.of equat', 'no. basic steps','basic time step'
Integration method of differential equations: Euler or 4-point Runge-Kutta. In case of the Runge-Kutta method, a fixed integration step can be specified (M-times smaller than printing step /basic time step/) or a variable integration step can be chosen which is adapted to the required accuracy 10**M.

INPUT 'Number of stimulus functions X', 'Index of TC as input X1', 'Index… X2',…
Number of inlet functions (time functions defined by data tables t-c)

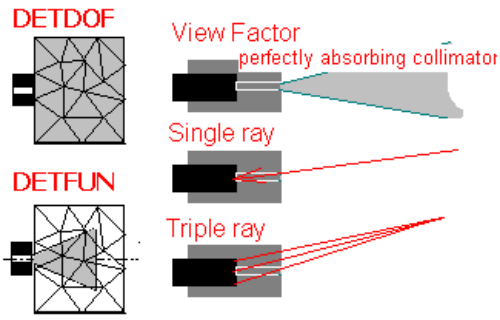
OUTPUT 'Number of response functions Y', 'Index of TC as response Y'
Number of outlet functions – recorded responses (time functions defined by tables)

3.15. Detectors

DETDEF (DD) *Detector number, x-front,x-back, y-front,y-back, z-front,z-back, aperture, view angle, range, attenuation*

definition of detector position using x,y,z coordinates of centreline points at the front and back of collimator hole. View angle and the range define a conical region, where integration is performed if the method DETFUN is chosen. Attenuation concerns material of collimator and this parameter is used only if the 1-ray or 3-rays algorithms of collimation are selected.

DETPLOT *Number of detectors* plots detectors
DETLIST *Number of detectors* list detectors



If you specify the last parameter Row=0 all time steps from OUT file will be processed, Row>0 processes only current distribution.

DETDOF (DETD) *dof (for example CN,CD, or CA), method (1-view factor, 2-view factor with near distance correction, 3,4-single ray without and with correction, 5,6-triple ray without and with correction, 7-detector without collimator), media attenuation, attenuation function, no.of detectors, detector 1, column of response 1, detector 2, column of response 2,…,row*

Responses of selected detectors at previously defined time steps (command T) are calculated by RUNDET.EXE integrating specified DOF over all elements. Parameter *media attenuation* is a constant multiplied by an *attenuation function* as soon as this function is defined. The command DETDOF calculates either responses at all time steps based upon results in file .OUT if the last parameter *row*=0 or just only one value, corresponding to the current distribution of tracer if *row*>0. In this second case results are transferred to the specified *row* of MOP.

DETFUN (DETF) *index of function, method (1-view factor, 2-view factor with near distance correction, 3,4-single ray without and with correction, 5,6-triple ray without and with correction, 7-detector without collimator), media attenuation, attenuation function, no.of detectors, detector 1, column of response 1, detector 2, column of response 2,…row*

Responses of selected detectors are calculated by integration of specified function within the specified region – a cone characterised by view angle and height (range of vision). The specified function can be usually DOF(CN,3,xx,yy,zz) returning concentration of tracer in the point xx,yy,zz.

There are three functions, which are useful when defining DETFUN and variable attenuation:

DET(detector,method,attenuation,x,y,z)	response i detector to point source at x,y,z
ALEN(detector,x,y,z,xc,yc,zc,a,b,h,ityp)	length of ray passing through a plate or cylinder
RLEN(detector,x,y,z)	distance of point x,y,z from detector.

3.16. Linear and nonlinear regression

Linear and nonlinear regression of data prepared in the matrix of observation points, using commands T, REOMAT and eventually TCF (see previous chapters). Independent variables X_i, corresponding measured values Y_i and estimated standard deviations σ_I can be located in arbitrary columns of MOP.

LINREG 'Degree of polynomial','Column X','Column Y','Column prediction','Column sigma'
Linear regression using polynomial $y=a_1+a_2x+...a_Nx^{N-1}$

NELREG 'Model (0-unknown,…)', 'Column X', 'Column Y', 'Col. prediction','Col. sigma'
Nonlinear regression $y=f(a,x)$ using the following models

1	2	3	4
$y = a_1 + a_2 e^{-a_3 x}$	$y = a_1 + a_2 e^{-a_4 x} + a_3 e^{-a_5 x}$	$y = a_1 + a_2 e^{-(\frac{x-a_3}{a_4})^2}$	$y = \frac{a_1 + a_2 x}{1 + a_3 x}$
5	6	7	8
$y = \frac{a_1}{1 + a_2 e^{-a_3 x}}$	$y = a_1 x^{a_3 - 1} e^{-a_2 x}$	$y = \frac{a_1}{\sqrt{x}} e^{-a_2 \frac{(x-a_3)^2}{x}}$	$y = \frac{a_1}{x\sqrt{x}} e^{-a_2 \frac{(x-a_3)^2}{x}}$
9	10	11	12
$y = a_1 + a_2 \sin(a_3 x + a_4)$	$y = a_1 + a_2 x + + a_3 \sin(a_4 x + a_5)$	$y = a_1 + a_2 \sin(a_4 x + a_5) + + a_3 \sin(a_6 x + a_7)$	$y = a_1 x^{a_2}$
13	14	15	16
$y = a_1 x^{a_4 - 1} \exp(-a_3 x) + + a_2 x^{a_6 - 1} \exp(-a_5 x)$	$y = a_1 + \frac{a_2}{\sqrt{x}} e^{-a_3 (a_4 - x)^2 / x}$	$y = a_1 + \frac{a_2}{x\sqrt{x}} e^{-a_3 (a_4 - x)^2 / x}$	$y = a_1 + a_2 x^{a_4 - 1} e^{-a_3 x}$

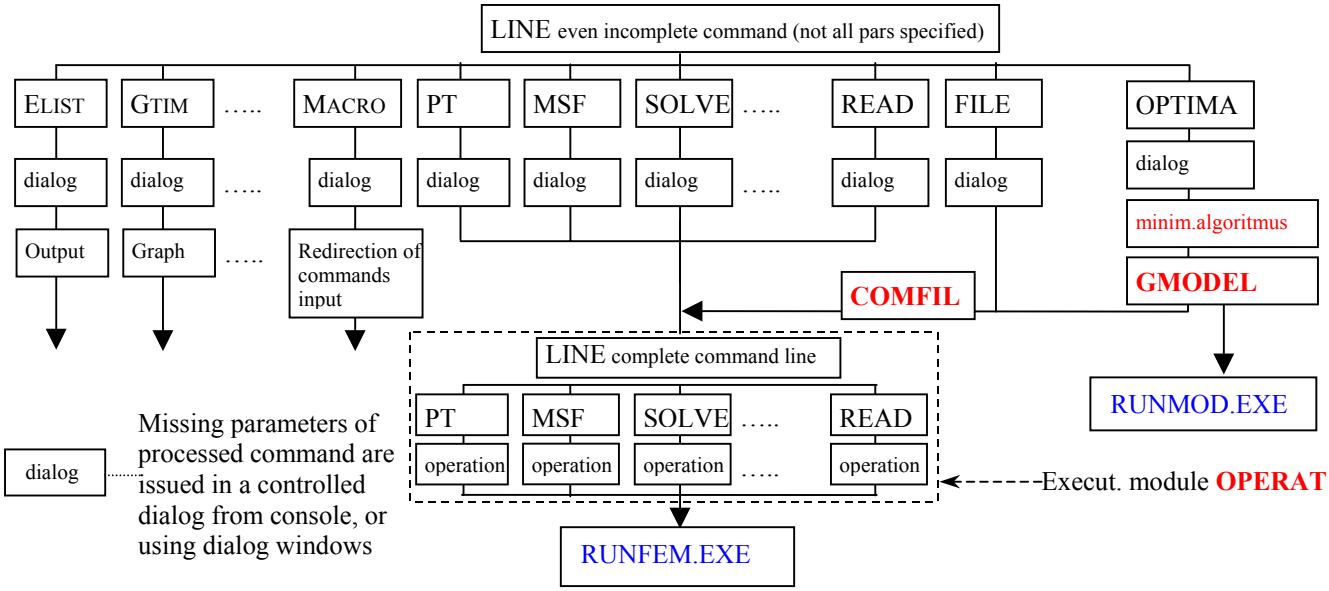
NLR2 'Model (0-unknown,…)', 'Column X1', 'Column X2', 'Column Y', 'Col. prediction','Col. sigma'
Nonlinear regression $y=f(a,x_1,x_2)$ using the following models with 2 independent variables

1	2
$y = a_1 + a_2 x_1 + a_3 x_2$	$y = a_1 + a_2 x_1 + a_3 x_2 + a_4 x_1 x_2$
3	4
$y = a_1 + a_2 x_1 + a_3 x_2 + a_4 x_1 x_2 + a_5 x_1^2 + a_6 x_2^2$	$y = a_1 + (a_2 + a_3 x_1 + a_4 x_2) \exp(a_5 x_1 + a_6 x_2)$
5	6
$y = a_1 x_1^{a_2} x_2^{a_3}$	$y = a_1 + a_2 x_1^{a_3} x_2^{a_4}$
7	
$y = a_1 + a_2 \frac{x_1^{a_3} x_2^{a_4}}{1 + a_5 x_1^{a_6} x_2^{a_7}}$	

3.17. Optimization of mathematical model

In the previous paragraphs (RTD models), the mathematical model describing a flow system by ordinary differential equations was introduced. The system of equations was written as a text which is interpreted during solution. The input is a vector of parameters which had to be defined before integration of the equation system, and the outputs are vectors describing system responses (in general N-responses with equidistant time step are stored in the observation points matrix). The concept of mathematical model can be extended to a more general form of a command file processed as a batch, which for a given vector of parameters calculates vectors of responses. Such a command file must be prepared in such a way, that its outputs are vectors in the observation matrix.

The way how a general mathematical model is implemented in FEMINA and how its parameters can be optimised is illustrated in the following scheme, describing modules for processing a general command line



You can see that there are two levels (layers, shells) of processing and some command lines are in fact processed twice, for example the command SOLVE if first completed in a controlled dialog, and this completed command line is processed again in the low layer module (OPERAT), i.e. even then the FEMINA database is written into a binary file as input data for program RUNFEM.EXE which is executed. On the other hand there are command parsed and executed only once on the top layer, e.g. the list command ELIST. The batch command FILE can be recognised also only on the top layer[10], and is processed by the module COMFIL (this module search for control commands like #LOOP in the command file and generates command lines which can be processed on the low level OPERAT).

The commands OPTIMA and SOMA are defined in the outer shell and activate optimisation of model parameters $p_1,\ldots,p_{NOPT}$ which should minimise the weighted sum of squares of deviation between experimental data in the matrix of observation points and the model prediction

$$s^2(p_1, p_2, ..., p_{NOPT}) = \sum_{j=1}^{L}\sum_{i=1}^{N}(y_{i,e_j} - y_{i,m_j})^2\, y_{i,w_j} \ .$$

where L is a number of compared response pairs, N is number of observation points (time steps) and $y_{i,j}$ is an element of the observation points matrix (this is default, another criteria of matching can be selected). The optimised model is represented by the subroutine GMODEL(ITYP,…), which carries out the following operations depending upon the value of ITYP:

[10] This is the reason, why you cannot use the command FILE as a part of command file processed as a batch.

ITYP=0, the system of differential equations described by an active interpreted model is solved, the δ functions are used as stimulus functions (so it is the impulse response model),

ITYP=1, the system of differential equations of active model is again solved but general stimulus functions, specified as data columns in the observation points matrix, are used instead of δ functions, initial conditions are zero (no matter if nonzero values are defined in //ini section).

ITYP=2, the system of ordinary dif.equations (active model) is solved taking into account stimulus functions as well as initial conditions prescribed in the //ini section.

ITYP>2, the file which is opened on the device number ITYP is interpreted (the computation is carried out by the COMFIL procedure similarly as the operation FILE *file*, described in the previous section). The file must opened by the command OPEN at first.

OPEN unit,file opens a batch file for interpretation as model on a selected unit

MOP file the same as OPEN, but always on unit 7, file name is selected from dialog panel.

CLOSE unit close the file opneden on specified unit.

Arguments of model (arguments of subroutine GMODEL) are parameters $p_1,\ldots,p_{NOPT}$, which can be any real or integer variables in the FEMINA database. The commands described in the preceding chapter (RTD models) serve for the parameters selections, initial values definition, specifications of allowable range and also whether and how these parameters are to be optimised:

PARDEF *No.of parameters,p_1-name,p_2-name,….*

PARSET $p_1, p_2,\ldots.$ set actual values of parameters.

PARLIM $p_{1min}, p_{1max}, p_{2min}, p_{2max},\ldots$ defines allowed range of parameters.

PARFIT p_1, *'0-supressed,1-linear,2-nonlin,3-search'*, ω_{p1}, p_2, *'0-supressed, ...'*, $\omega_{p2},\ldots$

The criterion of match between the model prediction and the reference data has been also described previously (see the chapter Time series processing)

CRITER *criterion*

COMPAR *Number of pairs, first row, last row, prediction$_1$, data$_1$, weight$_1$, prediction$_2$, ...*

TCDEV calculates and displays deviation for actual settings; result is in system variable SCOMP.

WMOD modification of weights in MOP.

There are two basic algorithms for optimisation model parameters, OPTIMA is deterministic and SOMA stochastic

OPTIMA 'Model 0,1,2 dif.eq., >2 file','Number of iteration', scaling, λ-increase, λ-decrease

The *Model* is either interpreted system of ordinary differential equations or a fully general model. *Model* 0,1 has to be already activated issuing the RMODEL command, other models must be prepared in the file which was opened by the OPEN command on an input device number *Model*.

SOMA 'Model 0,1,2 dif.eq., >2 file','Number of iteration','Specimen','PRT','step','mass'

Memetic optimisation algorithm SOMA, *Specimen* is number of specimen moving in a parameter space towards a leader (best specimen in previous iteration), PRT is perturbation of directional vector of specimen movement, STEP step size, MASS relative length of directional vectors (see chapter 4).

Typical sequence of commands follows

OPEN - opening a command file which can be processed as a batch

T - number and size of time steps used in simulation

READ - reading experimental data which are to be compared with model prediction (e.g. column 3)

COMPAR - defines position of prediction and experimental data in matrix of observation points

PARDEF - definition of model parameters (not only those which are subject of optimisation)

PARSET - initial values of model parameters (of all and not only optimised parameters)

PARLIM - specification of minimal and maximal values of optimised parameters,

PARFIT -

OPTIMA - execution of the optimization.

3.18. Command processing

Basically, three different methods of command specifications can be used in the FEMINA program. All the methods have been mentioned in previous sections so only basic characteristics and comparison follow:

- Interpretation of a command file as a batch is activated using the FILE command: Each line of the batch file represents one command which is interpreted by a special procedure and unknown commands are skipped. Commands must be specified with all necessary parameters (i.e., missing parameters cannot be added subsequently, for example point coordinates cannot be specified with mouse or by using keyboard). The procedure does not interpret commands like LIST, PLOT, GRAF designed for interactive work, but it interprets the commands for command processing control, e.g. #GOTO, #LOOP,…, and naturally all executive commands (entity definition, mesh generation, computations and even calling of external programs). Syntax of the control commands is like this:

  #GOTO *label* max. length of *label* is only 4 characters!
  #LOOP *label,n*
  body of cycle enclosed by #LOOP and #LABEL, executed *n*-times (*n* can be expression)
  #LABEL *label*
  #IF *expression*
  commands executed if *expression* is not zero (*expression* can be a relation, e.g. $a<2$)
  #ELSE
  commands executed if *expression* is zero
  #ENDIF

  The batch mode is necessary for definition of a model which is to be optimised – this action must be done automatically without operator intervention because the model is called many times.

Remark: There is an exception, the command VALUE *variable*, assigning a numerical value entered from keyboard to *variable*, will be also carried out in the batch mode. You have seen in previous examples that tasks are frequently composed of two or more batch files: the first one defining a fixed part of a problem makes use of the VALUE command for interactive setting of important parameters and even the next file defines the model which is to be executed several times without operator intervention.

- MACRO regime (command MACRO *name*) is seemingly similar: again, commands written in a text file are interpreted, but now the processed file need not represent a closed and fully defined problem because instead of an automatic processing only the keyboard input is redirected to the selected command file. This means that the control commands like jumps and loops #GOTO, #LOOP will not be executed but on the other hand the interactive commands like LIST, PLOT, GRAF…, will. What is important: all missing parameters of commands can be completed in a following dialog and therefore the macro file can be only a general prescription for solving specific problems (it can contain just only names of commands which should be executed). Special meaning is assigned to the lines beginning with C* (the whole line is a comment), Q* (processing of the macro is finished and standard interactive regime is resumed), R* (choice of previous command repetition, possibly with different parameters), I* commands which is to be inserted from keyboard (the I-line is available for editing in the edit panel activated by [Tab]).

- Interactive regime – commands are entered from keyboard and command parameters are specified in a following dialog. If the command keyword is not found within the standard FEMINA's commands, looking at the external program names list follows (the list is in the initialisation file

$FEMINA.CMD which is read at beginning of the FEMINA.EXE program run). When a match is found, a standard dialog specifying external program parameters begins (the dialog description is given in the file $FEMINA.CMD) and after that the external program is executed. If the first word of the command line is not a keyword, an attempt to interpret the command line as a command of the internal program interpreter is done (e.g. A=1.234) – that is after all the most natural and fastest way of specifying parameters. If an error in the script language is found by the interpreter, another possibility is checked – to use the keyword as a command of operating system MS-DOS®. The last possibility is a name of file which contains a macro: if a file of such a name (without postfix!) exists in the actual directory it will be interpreted. Only if even this last attempt fails the text „unrecognized command“ is displayed and the procedure HELP is called and similar commands are searched for (usually several variants are proposed).

3.19. Command interpreter language and system variables

Identifiers with maximal length of 8 characters are assigned to all variables and functions. The command VARLIST or [F2] reports the list of predefined variables with their descriptions. You can use the command TRANSE which displays actual number of system/user defined variables, and also the translated form of the last evaluated expression.

a) Scalar variables

The variables TIME,XX,YY,ZZ,TEMP,UX,UY,II,DP,RE,DE,TAU can be used as arguments of functions when defining boundary or initial conditions. Values 1,2,…,24 are assigned to variables TEMP, UX,…,EPS so that they can be used in a dialog of parameters specifying nodal parameters (DOF) with either name or number (e.g. the question: „Which nodal parameter do you want to display?“ can be replied as either number 1 or as the word TEMP – result is the same).

TIME, XX, YY, ZZ, II, DP, RE, DE, HE,TAU, TEMP, UX, UY, UZ, RX, RY, RZ, VOLT, VX, VY, VZ, PRES, OMG , PS, PSX , PSY, PSXX, PSYY, PSXY, CN, CD, CA, KT, EPS , FILM

Note: II-second invariant of deformation rate, DP-pressure drop, RE-Reynolds number, DE-equivalent diameter, HE-thickness, TAU-shear stress (II and TAU are used for definition of rheological model), TEMP-temperature, UX-displacement, RX-rotation, VOLT-voltage, VX-velocity, PRES-pressure, OMG-vorticity, PS-stream function,…

Element names (used similarly as the names of DOF)
PIPE2D, PIPE3D, TRUSS2D, TRUSS3D, BEAM2D, BEAM3D, CSTR, PUMP, VALVE, SHELLAX, HEXC, PLANE2D, FLOW2D, FLOW3D, FLOWT2D,FLOWT3D, SOLID, PSIN, PLINK, MASS

Names of element parameters
ENA, ENB, EMA, EMB, EQZ, EQ , ERE, ETAU, EFOUL, EDISS, ELAV, EII, ETX, ETY, ETZ, ESXX, ESYY, ESXY, EMISE , EPOWER, ETMEAN, EVX, EVY, EVZ, EALPHA

N_α, N_β, M_α, M_β, Q_z, Q-flowrate, Re, τ_w, k_{foul}, Δ:Δ, Length-Area-Volume, II, $T_{,x}$, $T_{,y}$, $T_{,z}$, σ_{xx}, σ_{yy}, σ_{xy}, σ_{Mises},…

Names of methods used for solution of Navier Stokes equations and names of heat exchangers
UVP,UVPP,MIKE,PENS,MIDE,PSIN,PSBL,CREE,PSOM,CARE,CHEV,STUB,HAIR

Scalar variables (REAL A,B,…, INTEGER I,J,…,N) set apart for user:
A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

System variables (INTEGER) for the finite element method
NPT, NCR, NSF, NVL, NE , ND, NEG, NRC, NMP, NTC

number of points, curves, surfaces, volumes, elements, nodes, groups EGROUP, RCONST and MPROP, number of time courses defined in matrix of observation points..

STEP, DT (REAL)

number of time or iterative steps, size of time step,
Example: DISP DT displays the value of time step

GX, GY, RUPW, SCL, LAMBDA, PIVOT, TOL, RELFAKT, INTEGRAL, THETA,SUMDEV, POWERF, POWERE, TAUD, BL, TREF (parameters REAL)

GX,GY acceleration in *x,y* directions, correction of standard upwind coefficient (RUPW), scale of drawing (SCL=0 automatic scale), penalty parameter λ (LAMBDA) of the pseudo-compressibility method, minimal size of pivot element (PIVOT), maximal distance of two nodes which can be merged during the merge operation (TOL), underrelaxation factor (RELFAKT), result of integration (INTEGRAL), time integration factor (THETA=1 implicit, THETA=0 explicit integration), deviation (SUMDEV), dissipated energy by friction (POWERF), electric power (POWERE), τ-parameter in the axial dispersion model, thermal boundary layer correction (BL), reference temperature (TREF).

Example: SCL=0.1 deformed structure will be displayed with increments UX,UY multiplied by 0.1.

CONV, BUOY, OHMI, UPW, HEPI, RT, NINTG, VOLFLOW, FAST (INTEGER flags and switches)

Convective terms (CONV=0/1), buoyancy forces, i.e. natural convection (BUOY=0/1-Boussineque/2-variable density), inner ohmic heating (OHMI=0/1), upwind modification (UPW=0/1), solution method (HEPI=0-weighted residuals/1-enthalpy balances/>1-method of characteristics), fouling models (RT=0/1-Novy/2-de Jong), number of integration points (NINTG), volumetric or mass balancing in pipes (VOLFLOW=0/1), fast evaluation of RTD models (FAST=1).

Example: CONV=0 suppress convective terms calculation
UPW=1 upwind activation

Integer parameters specifying number of iterations for a specific operation type which are performed in every time step of the finite element method solution

ELEC, THER, FLOW, CONC, STATIC

Example: ELEC=1 one iteration of electric field will be executed in time step

INTEGER parameters connected with models described by the system of ordinary differential equations (lumped parameter models):

NP

number of model parameters to be optimized (see also parameter vectors RP,IP)

METH, NEQ, NMOD

solution method of differential equations system (Euler, Runge-Kutta), NEQ is number of equations, NMOD is number of predefined interactive models.

NINP,NOUT

number of stimulus functions and number of model responses (see vectors XV,YV, VINP,VOUT),

b) Vectors

XPT(i), YPT(i), ZPT(i)

coordinates of keypoints,

Example: XPT(3)=(XPT(1)+XPT(2))/2

XND(i), YND(i), ZND(i)

node coordinates

IALG(i), RALG(i)

IALG and RALG specify an algorithm (each of these vectors has 50 elements). The most important elements of these vectors are identical with previously described scalar variables, e.g. IALG(2)≡CONV, IALG(3)≡BUOY,… IALG(11)≡ELEC, IALG(12)≡THER, IALG(13)≡CONC,... Zero values of the switches IALG usually block a specific part of computation: IALG(1)-computation of mass matrix, IALG(2)-forced convection, IALG(3)-natural convection, IALG(4)-source of heat, IALG(5)-upwind. Elements IALG(11) , IALG(12), … IALG(15) are numbers of permitted iterations in the scope of one time step with operations ELEC,FLOW,THER,CONC,STATIC.

Example: IALG(2)=0 ignoring of convective terms

RALG are parameters common for all elements: RALG(1) is equivalent to the scalar variable GX-g_x acceleration in the *x* direction, RALG(2) is equivalent to GY-g_y, RALG(3)-α heat transfer coefficient (this is not the only possibility of specifying the heat transfer coefficient, this parameter is also defined in RC zones of various material element groups, and it can be even specified separately in individual nodes as a nodal parameter), RALG(4)-T_e ambient temperature, RALG(5) equivalent to the variable RUPW-multiplies the coefficient of asymmetric test functions calculated according to the Peclet number of the element (zero value suppresses asymmetry of test functions - upwind), RALG(6)-required tolerance of pivot element, RALG(7)-relaxation factor, RALG(8) scale of increments when drawing deformed structure, RALG(9) penalty parameter λ, and so on.

Example: RALG(1)=9.81 sets the acceleration value
GX=9.81 completely the same.

Nodal parameters V1,V2,V3 and status of nodal parameters IPU are stored in the memory as matrices with rows corresponding to nodes and columns corresponding to the type of nodal parameter. The position and access to a particular nodal parameter is determined by the vector LPU

IPU(i), LPU(i), JPU(i)

status IPU of nodal parameter indexed by the pointer LPU, JPU-type of nodal parameter (TEMP,VX,…)

Example: DISP IPU(LPU(temp)+6) displays temperature status in node 6.
DISP JPU(2) type of parameter in column 2

V1(i), V2(i), V3(i)

vectors of nodal parameters indexed by the pointer LPU. V1 is the vector of entered values, V2 input and output of frontal method, V3 zone of initial values.

Example: V2(LPU(PRES)+I) is calculated pressure in the node I.

E1(i), E2(i), E3(i), E4(i), E5(i), E6(i)

vectors of element parameters (only the first six columns are available).

Example: E3(I) is the 3rd parameter value of element I (value in the 3rd column). Its meaning depends on the element type however it is possible to use standard mapping function IEP(name of parameter,element) which calculates index of specified parameter in the whole element parameter array.

HH(i), DD(i), PE(i), ALPHA(i), TE(i), AREA(i), PERIM(i), JZ(i), VOL(i), RA(i), DUDY(i), ZETA(i), PERM(i)

real constants *H, D, p* (pressure), α (heat transfer), T_e (temperature), *Area* (cross-section or heat transfer area), *O* (wetted perimeter of the cross-section), *V* (mixer/reactor volume), J_z (moment), relative roughness of wall (RA), transversal component of electric field intensity (dUdy), loss coefficient ζ, wall permeability. Index i indicates the group RCONST.

KX(i), CP(i), DENS(i), KAPPA(i), EX(i), MI(i), VISC(i), BETA(i), DN(i), HN(i), AN(i), TM(i), HM(i)

thermophysical parameters λ (thermal conductivity),c_p,ρ,κ (specific electric conductivity), *E* (modulus of elasticity), μ(Poisson constant), μ(viscosity), β(thermal expansion), D_N(diffusion coefficient), H_N(enthalpy of formation), A_N(frequency factor), T_M (melting point temperature), H_M (latent heat)

Example: VISC(2)=VISC(1) sets the same viscosity in the group 2 of material parameters as well as in the group 1.

ENAME(i), TRANS(i), GAUSS(i), AXIS(i), STRESS(i),TYPE(i), HYDR(i), HEAT(i), HSINK(i), CNSRC(i), CNSINK(i), CDSRC(i), CDSINK(i), CASRC(i), CASINK(i)

integer element parameters of groups EGROUP: ENAME-element type, TRANS-static 0/transient 1, GAUSS-number of integration nodes, AXIS-axis symmetry 1, STRESS-plane stress 1/plane deformation 0, TYPE – either method of NS-equations solution, type of pump or heat exchanger, HYDR – index of function defining hydraulic characteristics, HEAT, HSINK-index of functions defining volumetric heat source/sink, CNSRC, CNSINK – source/sink of component c_N and similarly for c_D, c_A.

Example: GAUSS(1)=3 sets number of Gauss integ. nodes in group 1

The part of FEMINA which solves systems of ordinary differential equations (lumped parameter models) uses vectors CM and DCM to describe systems of differential equations for tracer concentrations in basic model units – ideally mixed vessels:

CM(i), DCM(i)

concentrations and the first derivations with respect to time in the i-th mixer.

The system of ordinary differential equations has some parameters which are to be identified from experimental responses. The parameters can be arbitrary variables described in this chapter (e.g. A,B,…), however in the predefined models only the following vectors RP(i) and IP(i) are used:

RP(i), IP(i), PMIN(i), PMAX(i), PLOC(i), REGR(i),

model parameters RP and IP are identified vectors, the RP refers to *real*, and IP to *integer* values (model parameters can be real and integer). PMIN and PMAX are prescribed minimal and maximal limits of parameters, PLOC(i)

is position of i-th model parameter in the zone COMMON /FEM/, and the vector REGR determines parameters to be optimized.

VINP(i), VOUT(i), XV(i), YV(i)

integer vectors VINP and VOUT are indices of stimulus and responses functions. The index is the column of the matrix of observation points where a time course is located. XV(i) and YV(i) are vectors of current values of stimulus and response functions – these vectors are changing during the integration of differential equations.

C1(i), C2(i), C3(i), C4(i), C5(i), C6(i), C7(i), C8(i), C9(i), C10(i)

columns of the observation points matrix, the first column C1(i) is time and the other columns C2(i)… are corresponding values of time courses (which might be stimulus functions, model responses, experimental data, and even time history of nodal parameters values recorded during unsteady-state solution by the finite element method).

ITC(i), QTC(i)

vectors of additional information about columns of the observation points matrix (ITC-e.g. index of node, QTC-column type, e.g. *time*, *experiment*,…)

ICOMP(i)

indices of compared responses, selected columns of the observation matrix (number of compared pairs is NCOMP, calculated value of deviation according to the set criterion KCOMP is in the variable SCOMP).

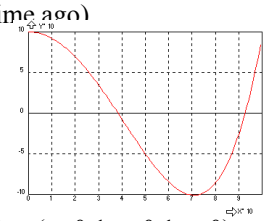
c) Functions

SIN(x) ,COS(x) ,EXP(x) ,SQR(x) ,LOG(x) ,LGT(x) ,ABS(x) , JN(n,x), YN(n,x),
MIN(a,b,c…) ,MAX(a,b,c,…) ,ATN(x) ,ERF(x) ,PLE(n,x) ,PLG(n,x) ,PTC(n,x), RND(x),

CVT(i,time), XVT(i,time), YVT(i,time),

DOF(name of nodal parameter, number of zone from 1 to 4, x,y,z) interpolated DOF
EPA(name of element parameter, x,y,z) interpolated element parameter
IEP(name of element parameter, index of element) index of parameter in the array E1(iep)
DET(index of detector,method,attenuation, x,y,z) point source response (theoretical)
ALEN(detector,x,y,z,xc,yc,zc,a,b,h,ityp) length of ray passing through a plate or cylinder
RLEN(detector,x,y,z) distance of point x,y,z from detector
CLEN(index of curve) length of curve

Example:	DISP min(10,5,12)	result is 5 (function with variable number of parameters)
	DISP LOG(EXP(1))	result is 1 (LOG is natural logarithm)
	DISP LGT(10)	result is 1 (such logarithm identifiers used HP long time ago)
	DISP ATN(1e20)	result is 1.57
	for i=1,4 do disp RND(0)	results 0.515, 0.398, 0.263, 0.744 (random numbers)
	PLE(n,x)	Legendre polynomial of n-order
	PLG(n,x)	Laguerre polynomial of n-order
	PLT(n,x)	Chebyshev polynomial,
	FUNDEF 1,PLT(4,XX)	and plot GRAFUN 1
	DISP DOF(TEMP,2,0.1,0.1,0.)	temperature interpolated in zone of results 2 at (x=0.1,y=0.1,z=0)
	DISP EPA(ETX,0.1,0.1,0.)	temperature gradient as element parameter at (x=0.1,y=0.1,z=0)
	for i=1,ne do E1(IEP(EFOUL,i))=0	zeroing fouling layer thickness in all NE elements.



Functions CVT,XVT,YVT return interpolated data in the observation points matrix. CVT(column, time), XVT(index of stimulus function, time), YVT(index of response, time).

Function DOF(dof,zone,x,y,z) is one of the most important: it returns the value of any nodal parameter in an arbitrary point *x,y,z* and this point need not be a node (special type of interpolation based on the distances of point *x,y,z* from all nodes in the element where the point is located is used). When the point *x,y,z* is located outside the finite element mesh, the returned value is 0.
The function EPA(epar,x,y,z), returning a value of specified element parameter, works similarly (element parameters are constant inside the element so that no interpolation is done and the whole problem reduces to finding the element which contains the point *x,y,z*).

Let us remark that both the functions DOF and EPA can be used only when the file $RUNBOX.BIN was created by the previous operation BOX.

The function DET(i,method,attenuation,x,y,z) returns the response of the detector *i* to the point source of radiation located at x,y,z (point source with unit activity of radiation). Response is calculated according to the selected method and according to parameters of detector defined by the command DETDEF. Parameter *attenuation* is average attenuation of media between detector and the point x,y,z.

d) Basic control commands identified by keywords

INT list, REAL list, Example: REAL ALFA,BETA(10)

WHILE condition DO block Example: WHILE I<10 DO [i=i+1 disp i]

Note: Block means a sequence of commands enclosed with brackets [ ] or the keywords *begin … end*. Only in the case that the block is represented by a single assignment or a DISP command the brackets can be omitted.

IF condition THEN block Example: IF IPU(LPU(TEMP)+i)>30 THEN V2(LPU(TEMP)+i)=1

IF condition THEN block ELSE block FOR i=1,NTS DO [IF i<50 THEN C2(I)=0 ELSE C2(I)=I]

FOR i=i1,i2 DO block Example:

```
FOR i=1,nd DO a=max(a,xnd(i))
FOR I=1,ND DO V3(LPU(VX)+I)=0
        zeroing X-component velocities
FOR I=1,ND DO XND(I)=XND(I)*0.1
        scaling coordinates of all nodes
```

DISP list Example: DISP NE,ND,NPT

RUN filename Example: RUN program.exe
RUN wordpad.exe

The command RUN enables to start external programs but it does not handle the problem of data transports. The operations WRITEB (writing database to a binary file), starting an external program and READB (reading results in a binary form) are joined together for a better user comfort and this combination actually represents the simplest possibility how to extend the program FEMINA with other operations. Names of these operations (keywords) and names of external programs are in the setup file $FEMINA.CMD which is read immediately after the FEMINA program start. The structure of the file is following

- Keyword (name of operation, e.g. TUPLEX – only first 4 characters are taken into account)
- Name of external program (e.g. TUPLEX.EXE)
- Number of parameters which are specified before the program start (same as with standard operations)
- Text corresponding to the first parameter
- default value of the first parameter
- Text corresponding to the second parameter

  …
- Keyword of another operation…

Example of the initialisation file:

```
\\extern
TUPL TUPLEX.EXE
Pipe diameter: 1.111
Pipe length: 2.222
Mass flowrate:3.333
Inlet temperature: 20
\\model
SERIES  mod1.txt
```

3.20. Alphabetical list of keywords and synonyms

#ELSE
#ENDIF
#GOTO
#IF
#LABEL
#LOOP
A,ACTNUM,NUMIN,NUM
ARC
B,INACTN,NUMOUT
BKCOLO
BOX,BOXING
BOXTES,INTERP
C,CR2PT,C2P,CRLINE,CURVE,KRIVKA
CIDENT,CID,PICKC,CRPICK
CIRCLE,CIR,KRUH
CLIST,CL,SETTIN,ENABLE
CLOSE
CLS,MAZ
CMD
COMPAR,TCOMP,TCOM,POROVN
CONVOL,CON
COPYTO,COTO,TCO
CR3PT,C3P
CRDEL
CRITER,TCRIT,TCRI
CRLIST,CRL
CRNPT,CNP
CRPLOT,CRP
CRSPOL,CRS,POLYLI,CRPOLY,CARA
CURDEF,CDEF,CD,DC,TABLE,TAB
CURLIS,CURL
D
DEBUG
DEFPLO
DETDEF,DD
DETDOF,DETD
DETFUN,DETF
DETLIS,DETL
DETPLO,DPL,DETPL
DISP
DISPC
DOFLIS,DOFL,RANGE
DSOLVE,SOLVED
E,ELEMEN
EDEL
EGLIST,EGL
EGROUP,EG
EIDENT,EID,PICKE,EPICK
ELIST,EL
ENDREC
EPLIST,EPL
EPLOT,EP
EPS
ERMEL
ERMOD
ERMSF
F,FILE,FDLG
FLUENF
FLUENT
FMOM
FOR
FUNDEF,FDEF,FD,DF,FUNKCE
FUNLIS,FUNL
G,ACTSET,SETGRP,ACTGRP,GROUP
GCR,GC,GDCR,GDC
GD1,G1,D1,GRAF1
GD3,GFSF
GE2,E2
GFCR,GCF,GFC
GRAFUN,GF
GRAPH,GD2,G2,D2,GRAF2,VRSTEV
GRAPS,GE1,E1
GRATIM,GT,GTIM,GTIME,HISTOR
GRIDOF,GOF
GRIDON,GRID,GON,MRIZKA
GTC,GXY
GTD
GTE
GVEL,GV
H,HELP,?
I,INITIA,INI,NULUJ
IDMSER,IDM,SERIE
IF
IMPULS,IMP
INIFEM
INIMOD,IMOD,IM
INLET,VSTUP
INPUT,INP,STIMUL,VZRUCH
INTGCR,ICR,IC
INTGSF,ISF,IS
INTGEL,INTEL,IE
J,TCL,TCLIST
K,ANALYS,ANAL,AN
L,LOC,WHERE,VAR,VARIAB
LA
LINREG,LR
LOADTE,LTE
LOADTN,LTN,LT,LOADT,CTICAS
M,MACRO,MAC
MAGNIF,MA,MG
MCR,MCRC,M_CR,MESHCR,SITC
MCR2,MCRPH,MCR24
MCR4,MCRHEX,MCH
METHOD,MET,RUNGE
MODLIS,MODL,ML
MOFE
MOMENT,MOM
MPLIST,MPL
MPROP,MP,MATERI
MSF,M_SF,MESHSF,SITS
MVL,M_VL,MESHVL,SITV
N,NEWPRO,NEW,RESET,START
NAMEPR,NAME
ND,NODE,UZEL
NDEL,NDDEL
NDLIST,NDL,NLIST,Y
NDPLOT,NDP
NELREG,NLR
NF,FN,NDF,FND
NFCR,FCR,BOUND,PODMIN
NFLIST,NFL,FLIST
NFPLOT,NFP
NFPT,FPT,FP,PF,PTF
NFSF,FSF,SOURCE
NIDENT,NID,PICKN,NDPICK
NLR2
NMERGE,NM,SLOUCI
NORM,NOR
O,OUTPUT,OUT,OUTLET,RESPON,ODEZVA
OPEN
OPTIMA,OPT,MINSSQ
OPTION,ATRANE,APIPE
P,PT,POINT,BOD
PALIS,PL
PARDEF,PARD,PADEF,PAD,PDEF
PARDEL
PARFIT,PARREG,PARF,PAF
PARLIM,PARL,PALIM,PLIM
PARSET,PARS,PASET,PAS,PSET
PASERI,PASER,SERIE2
PD,PGDN
PFPLOT,PFP,DIAL,METER
PIDENT,PID,PICKP,PTPICK
PMERGE,PM
PPT,
PSOLVE,SOLVEP
PTCRIN,PTCR2,PCI
PTDEL
PTLIST,PTL,PLIST
PTPLOT,PTP
PU,PGUP
Q,EXIT,QUIT,KONEC
RCEHEV
RCEPUM,RCEP
RCESTH
RCESTU,RCEST
RCLIST,RCL
RCONST,RC
RCPUMP,RCP
RCROSS,RCR
RCSTD
RCSTH,RCSTHEL
RCSTUB,RCST
READ,R,RTC,RNOD,READTC,READCO
READBI,READB,RB
READTS,RTS
RECORD,REC
REM
REOMAT,ROM
RCHEV,RCHEVR
RCHEX,RCHE
RMODEL,RM,MODEL
RUN
S,STATUS,SHOW
SAVE,ULOZ
SEARCH
SETWIN
SF3PT,S3P
SF4PT,S4P,SURFAC,PLOCHA
SF6PT,S6P
SF8PT,S8P
SFCR
SFDEL
SFEXTR,SFEX
SFLIST,SFL,SLIST
SFPLOT,SFP
SMOOTH,SM,TCSMOO,TCSM,SPLINE
SOLVE
SOMA
T,TIMES,TSTEP,DT,CAS
TAIL,CHVOST
TCBGR,BGR,POZADI
TCDEV,TCDELT,ODCHYL
TCDT,TCT
TCEDIT,TED,TCE
TCEND,TCRANG
TCFFT,FFT,FT
TCFR
TCFUN,TCF,FUNTC
TCINP,TCI,INTC,INPUTC
TCINT,TCINTG
TCJOIN,TCJ,UNION,COMPOS
TCLIP,CLIP
TCOSHF,TCOS,OS
TCPLOT,TCP
TCPXY,XY
TCREG,TCR
TCRND,RND,RAND,SUM
TCSET,SETC,STC,TCTYPE,TCS
TCSUB,SUBTRA,SUB
TCXSHF,TSHIFT,TS
TCYSHF,SHIFT,CS
TCZERI,TCZ,TCZERO,TCZER
TIDENT,TID,PICKT,TCPICK
TRANSE
TRUSS,TRUS
U
UNDO,BACK,ZPET
UNIFIL,IMPORT
V,VARLIS,VARL
VALUE,ASSIGN
VIEW,V3D,POHLED
VL8PT,V8P,OBJEM
VLLIST,VLL,VLIST
VLPLOT,VLP
VLSF
WALL,STENA
WHILE
WINDOW,OKNO
WMODIF,WMOD
WRITE,W

The most frequently used commands activated by single letter

A-actnum (numbering entities in graph)
B-iactnum (deactivation of numbering)
C-CR2PT (line)
D-Down (display scroll)
E-Element defined from nodes
F-File (interpret of batch)
G-group (group activation)
H-Help
I-Initia (initial conditions)
J-TCLIST (observation points matrix)
K-type of analysis
L-LOCate (variable)
M-interpret file as a macro
N-NEW
O-Output
P-PT (point definition)
Q-display of settings
R-read
S-show (status window)
T-time steps
U-Up (display scroll)
V-varlist (list of variables)
W-write
X-list of nodes and DOF
Y-list of element parameters
Z-definition of z-coordinate

Alphabetical list of system variables

```
FUNCTIONS ---------------------------------------------

ALEN(detector,x,y,z,xc,yc,zc,a,b,h,ityp)
ATN(x)
CVT(column,time)   : interpolate MOP
DET(detector,method,attenuation,x,y,z)   point response
DOF(dof,zone,x,y,z)
EPA(parameter,x,y,z)
ERF(x)         : error function erf(x)
JN(n,x)        : Bessel function of the first kind Jn(x)
MAX(x1,x2,...) MIN(x1,x2,...)
PLE(n,x)       : Legendre polynomial
PLG(n,x)       : Laguerre polynomial
PTC(n,x)       : Tchebyshev polynomial
RLEN(detector,x,y,z)
RND(seed)
XVT(i,time)    : interpolated stimulus function
YVT(i,time)    : interpolated response function
YN(n,x)        : Bessel function of the second kind Yn(x)

SIMPLE VARIABLES INTEGER ------------------------------

BUOY :buoyancy (1-density, 2-Boussinesque
CONC :max.iterations CONC (2D concentration)
CONV :convective terms respected
ELEC :max.iterations ELEC (2D voltage)
ENDC :end of COMMON/$FEM/ zone (only for checking)
FLOW :max.iterations FLOW (Navier Stokes)
HEPI :=0 MWR, =1 enthalpy balance, >1 characteristics
I    :user variable I-integer (can be used for anything)
J    :user variable J-integer (can be used for anything)
K    :user variable K-integer (can be used for anything)
KCOMP:criterion for TC comparison 0-sum,...4-integral...
L    :user variable L-integer (can be used for anything)
M    :user variable M-integer (can be used for anything)
METH :solution of eqs. by 0-Euler,1-Runge Kutta
N    :user variable N-integer (can be used for anything)
NCOMP:number of pairs of compared TC curves
NCR  :number of curves created by using CR2PT,CR3PT
ND   :number of nodes (created by MCR,MSF,... or by ND)
NE   :number of finite elements (created by MCR,MSF,...)
NEG  :number of E-groups (defined by EG command)
NEQ  :number of equations and unknowns cm(1),...cm(neq)
NINP :number of stimulus functions xvt(i,t) for a model
NINTG:number of integration points along curve
NMOD :number of interpreted models defined in ext.files
NMP  :number of MPROP-groups (defined by MP command)
NOUT :number of recorded responses yvt(i,t) of model
NP   :number of lumped model parameters PM(1),...
NPART :number of particles (represented by finite elements SPHERE,…)
NPT  :number of points created by PT command (geometry)
NRC  :number of RC-groups (defined by RC command)
NSF  :number of surfaces created from points or curves
NSGAP :switch on solution of Navier Stokes equation in a gap
NT   :number of points of a curve (no.of time steps)
NTC  :number of points of a curve (no.of time steps)
NVL  :number of volumes (volumes are not supported yet)
OHMI :source terms respected
PHASE:phase changes model
RT   :Fouling model
STATIC  :max.iterations STAT (stress analysis)
```

```
STEP :number of time steps
THER :max.iterations THER (2D thermal)
UPW  :upwind enabled
VOLFLOW :volumetric/mass balancing in pipes

VECTORS(i) INTEGER ----------------------------------------

AXIS  :EGROUP   Cartesian/Cylindrical coord.syst. assumed
CASINK:Ca sink function
CASRC :Ca source function
CDSINK:Cd sink function
CDSRC :Cd source function
CNSINK:Cn sink function
CNSRC :Cn source function
ENAME :element name
GAUSS :EGROUP   number of Gauss Points
HEAT  :heat source function
HSINK :heat sink function
HYDR  :hydraulic characteristic function
IALG  :switches (MASS-matrix,CONVECTion,BUOYANCY,SOURCe)
ICOMP :indices of compared curves...
IP    :lumped model parameters
IPU   :IPU(lpu(i)+j) status of DOF (<0 fixed, >0 load)
ITC   :correponding node(1),exper.(2),inlet(3),outlet(4)
JPU   :JPU(i) type of DOF (1-TEMP,2-UX,...)
LPU
PLOC  :association of parameter PM to a variable in FEM
QTC   :time course: 1-FEM,2-experiment,3-inlet,4-outlet
REGR  :regression parameter identification (=1)
STRESS
TRANS :EGROUP   Static/Transient algorithms
TYPE
VINP  :index of curve describing i-th stimulus xvt(i,t)
VOUT  :index of curve describing i-th response yvt(i,t)

SIMPLE VARIABLES REAL -----------------------------------

A     :user variables A,B,...H-real, I,J,...N-integer
B     :user variable B-real (can be used for anything)
BEAM2D:Element identification by name (BEAM2D)
BEAM3D:Element identification (not used)
BL    :thermal boundary layer rediction coef.
C     :user variable C-real (can be used for anything)
CA    :DOF identification as concentration CA (value 23)
CARE  :flow solution identif.(PSI-OMEGA Campion Rens.)
CD    :DOF identification as concentration CD (value 22)
CN    :DOF identification as concentration CN (value 21)
CREE  :flow solution identif.(stream function creep)
CSTR  :Element identification (CSTR - mixed tank)
D     :user variable D-real (can be used for anything)
DE    :second invariant of rate od deformation
DP    :pressure drop
DT    :time step [s]
E     :user variable E-real (can be used for anything)
EALPHA:E-par.identification (Ealpha)
EDISS :E-par.identification (dissip.power)
EFOUL :E-par.identification (fouling layer)
EII   :E-par.identification (second invariant)
ELAV  :E-par.identification (Length-Area-Volume)
EMA   :E-par.identification (Malpha meridian moment)
EMB   :E-par.identification (Mbeta circumf.moment)
EMISE :E-par.identification (Sigma Mises)
ENA   :E-par.identification (Nalpha normal stress)
ENB   :E-par.identification (Nbeta circumf.stress)
```

```
EPOWER:E-par.identification (EPOWER)
EQ    :E-par.identification (flowrate)
EQZ   :E-par.identification (trans.force)
ERE   :E-par.identification (Reynolds number)
ESXX  :E-par.identification (Sigma xx)
ESXY  :E-par.identification (Sigma xy)
ESYY  :E-par.identification (Sigma yy)
ETAU  :E-par.identification (wall shear stress)
ETMEAN:E-par.identification (mean temperature)
ETX   :E-par.identification (dT/dx)
ETY   :E-par.identification (dT/dy)
ETZ   :E-par.identification (dT/dz)
EVX   :E-par.identification (Vx velocity)
EVY   :E-par.identification (Vy velocity)
EVZ   :E-par.identification (Vz velocity)
F     :user variable F-real (can be used for anything)
FLOW2D:Element identification (FLOW2D -laminar flow)
FLOW3D:Element identification (FLOW3D -laminar flow)
FLOWT3D :Element identification (not used)
G     :user variable G-real (can be used for anything)
GX    :GX-gravity acceleration (effects natural convec.)
GY    :GY-gravity acceleration (effects natural convec.)
H     :user variable H-real (can be used for anything)
HAIR  :heat exchanger identification (hair)
HE
HEXC  :Element identification (HEXC heat exchanger)
HT    :DOF identification as enthalpy HT (value 24)
CHEV  :heat exchanger identification (plate-chevron)
II    :second invariant of rate od deformation
INTEGRAL:result of integration
LAMBDA:LAMBDA-penalty factor (should be 10**7 or so)
MASS  :Element identification (not used)
MIDE  :flow solution identif.(minimum dissip.energy)
MIKE  :flow solution identif.(minimum kinetic energy)
O     :user variable O-real (can be used for anything)
OMG   :DOF identification as vorticity omega (value 13)
OMGX  :DOF vorticity derivative OMGX (value 18)
OMGY  :DOF vorticity derivative OMGY (value 19)
OMGZ  :DOF vorticity derivative OMGZ (value 20)
P     :user variable P-real (can be used for anything)
PENS  :flow solution identif.(penalty method)
PIPE2D:Element identification by name (PIPE2D)
PIPE3D:Element identification (not used)
PIVOT :minimum of pivot (frontal method); default 1e-6
PLANE2D:Element identification (PLANE2D plate-membrane)
PLINK :Element identification (not used)
POWERE:electric power
POWERF:dissipated power by viscous friction
PRES  :DOF identification as pressure (value 12)
PS    :DOF identification as stream function (value 14)
PSBL  :flow solution identif.(stream function cubic)
PSIN  :flow solution identif.(stream function PSI)
PSOM  :flow solution identif.(PSI-OMEGA)
PSWALL:value of stream function at wall (for WALL)
PSX   :DOF stream function derivative PSX (value 15)
PSY   :DOF stream function derivative PSY (value 16)
PSZ   :DOF stream function derivative PSZ (value 17)
PUMP  :Element identification (PUMP)
Q     :user variable Q-real (can be used for anything)
R     :user variable R-real (can be used for anything)
RE    :Reynolds number
RELFAKT:Underrelaxation factor (iterations loops)
RUPW  :upwind correction (a value between 0 and 1)
RX    :DOF identification as RX rotation (value 5)
```

```
RY    :DOF identification as RY rotation (value 6)
RZ    :DOF identification as RZ rotation (value 7)
S     :user variable S-real (can be used for anything)
SCL   :scale factor for displacement (graphics)
SCOMP :norm of TC differences
SHELLAX:Element identification by name (SHELLAX)
SOLID :Element identification (SOLID- 3D brick)
STUB  :heat exchanger identification (shell and tube)
SUMDEV:sum od deviations
T     :user variable T-real (can be used for anything)
TAU   :second invariant of rate od deformation
TAUD  :characteristic time (axial dispersion)
TEMP  :DOF identification as temperature (value 1)
THETA :0-explicit,0.5-Crank Nicholson,1-implicit
TIME  :current time (transient analysis)
TOL   :minimum distance of points or nodes for merging
TREF  :reference temperature
TRUSS2D:Element identification by name (TRUSS2D)
TRUSS3D:Element identification (not used)
U     :user variable U-real (can be used for anything)
USER  :Element identification by name (USER defined)
UVP   :flow solution identif.(velocity-pressure)
UVPP  :flow solution identif.(pseudo-compressibility)
UX    :DOF identification as UX displacement (value 2)
UY    :DOF identification as UY displacement (value 3)
UZ    :DOF identification as UZ displacement (value 4)
V     :user variable V-real (can be used for anything)
VALVE :Element identification (VALVE)
VOLT  :DOF identification as voltage (value 8)
VX    :DOF identification as VX velocity (value 9)
VY    :DOF identification as VY velocity (value 10)
VZ    :DOF identification as VZ velocity (value 11)
W     :user variable W-real (can be used for anything)
X     :user variable X-real (can be used for anything)
XX,YY,ZZ:current coordinate (set before funct.evaluation)
Y     :user variable Y-real (can be used for anything)
Z     :user variable Z-real (can be used for anything)

VECTORS(i) REAL ----------------------------------------

ALPHA :RCONST   [W/m^2/K] heat transfer coeffient
ALPHAE:RCONST   electric conductivity of wall
AN    :MPROP    An frequency factor dc/dt=An*exp(En/RT)*c
AREA  :RCONST   [m^2] -usually heat transfer surface
BETA  :MPROP    Beta [1/K] thermal expansion
C1    :time values of curve (nodal history,experiment...)
C2,C3,...,C10  :values of curve (nodal history, experiment,...)
CM    :C(i) concentration in i-th vessel (lump.par.model)
CP    :MPROP    Cp [J/kg/K] heat capacity
DCM   :DC(i) time derivative of C(i)
DD    :RCONST   D [m] diameter of a pipe
DENS  :MPROP    Rho [kg/m^3] density
DEV   :Abs.values of DOF differences (iterat.termination)
DN    :MPROP    Dn [m^2/s] diffusion coeffient
DUDY  :RCONST   transversal intensity of el.field
E1    :first parameter of i-th element (postproc.result)
E2    :second parameter of i-th element (postproc.result)
E3    :third parameter of i-th element (postproc.result)
E4    :fourth parameter of i-th element (postproc.result)
E5    :fifth parameter of i-th element (postproc.result)
E6    :sixth parameter of i-th element
EX    :MPROP    E [Pa] (Young modulus)
HH    :RCONST   H [m]  thickness of a plate
HM    :MPROP    Hm [J/kg] enthalpy of melting
```

```
HN    :MPROP    En activation energy (Arrhenius)
JZ
KAPPA :MPROP    Kappa [S/m] electrical conductivity
KAPPAF:MPROP    [S/m] conduct.of fouling layer
KX    :MPROP    Kx [W/m/K] heat conductivity
KXF   :MPROP    [W/m/K] therm.cond.of fouling layer
MEAN  :Mean values of DOF's (calculated by RANGE)
MI    :MPROP    Mi (Poisson constant)
PE    :RCONST   Inner pressure [Pa] applied to a shell
PERIM :RCONST   Perimeter [m] of a cross section
PERM  :RCONST   permeability of wall
PMAX  :lumped model parameters upper bounds
PMIN  :lumped model parameters lower bounds
RA    :RCONST   wall rougness
RALG  :GX-gravity acceleration (effects natural convec.)
RNG   :DOF-Minimum values of all DOFs (command RANGE)
RP    :lumped model parameters
TE    :RCONST   Te [C] ambient temperature
TM    :MPROP    Tm [C] melting point temperature
UE    :RCONST   external voltage
V1    :V1(lpu(j)+i) DOF -j- of i-th node (input values)
V2    :V2(lpu(j)+i) DOF -j- of i-th node (results)
V3    :V1(lpu(j)+i) DOF -j- of i-th node (initial cond.)
VISC  :MPROP    Viscosity [Pa.s]
VOL   :RCONST   Volume
XND   :nodal coordinates X
XPT   :keypoint coordinate x
XV    :actual value of i-th stimulus functions xv(i)
YND   :nodal coordinates Y
YPT   :keypoint coordinate y
YV    :actual value of i-th response functions yv(i)
ZETA  :RCONST   local loss coefficient
ZND   :nodal coordinates Z
ZPT   :keypoint coordinate Z
```