LS-DYNA®
KEYWORD USER'S MANUAL

VOLUME III

Multi-Physics Solvers

12/01/22 (r:15373)
LS-DYNA Dev

ANSYS

Websites

https://lsdyna.ansys.com
https://support.ansys.com

Disclaimer

TABLE OF CONTENTS

TABLE OF CONTENTS 0-3

INTRODUCTION 1-1

*BATTERY 2-1
- *BATTERY_DATABASE_HISTORY_ALLNDS_ON_ELESET 2-2
- *BATTERY_DATABASE_HISTORY_NODELIST_ON_ELESET 2-3
- *BATTERY_DATABASE_HISTORY_GLOBALS 2-5
- *BATTERY_ECHEM_CELL_GEOMETRY 2-6
- *BATTERY_ECHEM_CONTROL_SOLVER 2-8
- *BATTERY_ECHEM_INITIAL 2-15
- *BATTERY_ECHEM_MAT_ANODE 2-16
- *BATTERY_ECHEM_MAT_CATHODE 2-18
- *BATTERY_ECHEM_MAT_ELECTROLYTE 2-20
- *BATTERY_ECHEM_THERMAL 2-22

*CESE 3-1
- *CESE_BOUNDARY_AXISYMMETRIC 3-3
- *CESE_BOUNDARY_BLAST_LOAD 3-6
- *CESE_BOUNDARY_CONJ_HEAT 3-9
- *CESE_BOUNDARY_CYCLIC 3-12
- *CESE_BOUNDARY_FSI 3-18
- *CESE_BOUNDARY_NON_REFLECTIVE 3-21
- *CESE_BOUNDARY_PRESCRIBED 3-24
- *CESE_BOUNDARY_PRESCRIBED_VN 3-28
- *CESE_BOUNDARY_REFLECTIVE 3-32
- *CESE_BOUNDARY_SLIDING 3-34
- *CESE_BOUNDARY_SOLID_WALL 3-36
- *CESE_CHEMISTRY_D3PLOT 3-40
- *CESE_CONTROL_LIMITER 3-41
- *CESE_CONTROL_MESH_MOV 3-42
- *CESE_CONTROL_SOLVER 3-43
- *CESE_CONTROL_TIMESTEP 3-45
- *CESE_DATABASE_ELOUT 3-46
- *CESE_DATABASE_FLUXAVG 3-48

TABLE OF CONTENTS

*CESE_DATABASE_FSIDRAG 3-50
*CESE_DATABASE_POINTOUT 3-51
*CESE_DATABASE_SSETDRAG 3-53
*CESE_DEFINE_NONINERTIAL 3-55
*CESE_DEFINE_POINT 3-57
*CESE_DRAG 3-58
*CESE_EOS_CAV_HOMOG_EQUILIB 3-59
*CESE_EOS_IDEAL_GAS 3-60
*CESE_EOS_INFLATOR1 3-61
*CESE_EOS_INFLATOR2 3-63
*CESE_FSI_EXCLUDE 3-66
*CESE_INITIAL 3-67
*CESE_INITIAL 3-68
*CESE_INITIAL_CHEMISTRY 3-69
*CESE_INITIAL_CHEMISTRY_ELEMENT 3-71
*CESE_INITIAL_CHEMISTRY_PART 3-73
*CESE_INITIAL_CHEMISTRY_SET 3-75
*CESE_MAT_000 3-77
*CESE_MAT_001(_GAS) 3-78
*CESE_MAT_002 3-79
*CESE_PART 3-81
*CESE_SURFACE_MECHSSID_D3PLOT 3-82
*CESE_SURFACE_MECHVARS_D3PLOT 3-83

*CHEMISTRY 4-1
*CHEMISTRY_BATTERY 4-3
*CHEMISTRY_COMPOSITION 4-4
*CHEMISTRY_CONTROL_0D 4-5
*CHEMISTRY_CONTROL_1D 4-7
*CHEMISTRY_CONTROL_CSP 4-9
*CHEMISTRY_CONTROL_FULL 4-10
*CHEMISTRY_CONTROL_INFLATOR 4-11
*CHEMISTRY_CONTROL_TBX 4-14
*CHEMISTRY_CONTROL_ZND 4-15
*CHEMISTRY_DET_INITIATION 4-16
*CHEMISTRY_INFLATOR_PROPERTIES 4-18
*CHEMISTRY_MODEL 4-23

*CHEMISTRY_PATH 4-25

*DUALCESE 5-1
*DUALCESE_BOUNDARY_AXISYMMETRIC 5-7
*DUALCESE_BOUNDARY_CYCLIC 5-9
*DUALCESE_BOUNDARY_FSI 5-13
*DUALCESE_BOUNDARY_NON_REFLECTIVE 5-16
*DUALCESE_BOUNDARY_PRESCRIBED 5-18
*DUALCESE_BOUNDARY_PRESCRIBED_HYBRID 5-23
*DUALCESE_BOUNDARY_PRESCRIBED_PHASE_CHANGE 5-30
*DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE 5-36
*DUALCESE_BOUNDARY_PRESCRIBED_VN 5-41
*DUALCESE_BOUNDARY_REFLECTIVE 5-45
*DUALCESE_BOUNDARY_SOLID_WALL 5-47
*DUALCESE_CONTROL_LIMITER 5-52
*DUALCESE_CONTROL_MESH_MOV 5-53
*DUALCESE_CONTROL_SOLVER 5-54
*DUALCESE_CONTROL_TIMESTEP 5-56
*DUALCESE_DATABASE_HISTORY_ELEMENT_SET 5-57
*DUALCESE_DATABASE_HISTORY_GLOBALS 5-58
*DUALCESE_DATABASE_HISTORY_NODE_SET 5-59
*DUALCESE_DATABASE_HISTORY_POINT_SET 5-60
*DUALCESE_DATABASE_HISTORY_SEGMENT_SET 5-61
*DUALCESE_D3PLOT 5-63
*DUALCESE_D3PLOT_FLUID_SSID 5-65
*DUALCESE_ELE2D 5-67
*DUALCESE_ELE3D 5-68
*DUALCESE_ELEMENTSET 5-70
*DUALCESE_EOS_CAV_HOMOG_EQUILIB 5-71
*DUALCESE_EOS_COCHRAN_CHAN 5-72
*DUALCESE_EOS_COOLPROP 5-74
*DUALCESE_EOS_IDEAL_GAS 5-78
*DUALCESE_EOS_INFLATOR1 5-79
*DUALCESE_EOS_INFLATOR2 5-81
*DUALCESE_EOS_JWL 5-84
*DUALCESE_EOS_NASG 5-86
*DUALCESE_EOS_REFPROP 5-88

TABLE OF CONTENTS

*DUALCESE_EOS_REFPROP_PATH 5-92
*DUALCESE_EOS_SET 5-93
*DUALCESE_EOS_STIFFENED_GAS 5-94
*DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED 5-95
*DUALCESE_FSI_EXCLUDE 5-97
*DUALCESE_INCLUDE_MODEL 5-98
*DUALCESE_INITIAL 5-99
*DUALCESE_INITIAL_SET 5-100
*DUALCESE_INITIAL_HYBRID 5-102
*DUALCESE_INITIAL_HYBRID_SET 5-104
*DUALCESE_INITIAL_PHASE_CHANGE 5-106
*DUALCESE_INITIAL_PHASE_CHANGE_SET 5-108
*DUALCESE_INITIAL_TWO-PHASE 5-110
*DUALCESE_INITIAL_TWO-PHASE_SET 5-112
*DUALCESE_MAT_GAS 5-114
*DUALCESE_MAT_GAS_0 5-116
*DUALCESE_MAT_GAS_2 5-117
*DUALCESE_MESH_GEOMETRY 5-119
*DUALCESE_MESH_PART 5-121
*DUALCESE_MODEL 5-122
*DUALCESE_NODE2D 5-123
*DUALCESE_NODE3D 5-124
*DUALCESE_NODESET 5-125
*DUALCESE_PART 5-126
*DUALCESE_PART_MULTIPHASE 5-128
*DUALCESE_POINTSET 5-130
*DUALCESE_REACTION_RATE_IG 5-131
*DUALCESE_REACTION_RATE_IG_REDUCED 5-133
*DUALCESE_REACTION_RATE_P_DEPEND 5-134
*DUALCESE_SEGMENTSET 5-135
*DUALCESE_SOLVER_SELECTION 5-137
*DUALCESE_SOLVER_CAV_EQNS 5-138
*DUALCESE_SOLVER_EULER_EQNS 5-139
*DUALCESE_SOLVER_HYBRID_MULTIPHASE 5-140
*DUALCESE_SOLVER_NAVIER_STOKES 5-142
*DUALCESE_SOLVER_PHASE_CHANGE 5-143
*DUALCESE_SOLVER_TWO-PHASE_MULTIPHASE 5-144

*EM 6-1
*EM_2DAXI 6-5
*EM_BOUNDARY 6-6
*EM_BOUNDARY_PRESCRIBED 6-7
*EM_CIRCUIT 6-9
*EM_CIRCUIT_CONNECT 6-14
*EM_CIRCUIT_ROGO 6-15
*EM_CONTACT 6-16
*EM_CONTACT_RESISTANCE 6-18
*EM_CONTACT_SUBDOM 6-21
*EM_CONTROL 6-23
*EM_CONTROL_CONTACT 6-25
*EM_CONTROL_COUPLING 6-27
*EM_CONTROL_EROSION 6-29
*EM_CONTROL_MAGNET 6-30
*EM_CONTROL_SOLUTION 6-31
*EM_CONTROL_SWITCH 6-33
*EM_CONTROL_SWITCH_CONTACT 6-34
*EM_CONTROL_TIMESTEP 6-35
*EM_DATABASE_CIRCUIT 6-37
*EM_DATABASE_CIRCUIT0D 6-39
*EM_DATABASE_ELOUT 6-40
*EM_DATABASE_FIELDLINE 6-41
*EM_DATABASE_GLOBALENERGY 6-44
*EM_DATABASE_NODOUT 6-45
*EM_DATABASE_PARTDATA 6-46
*EM_DATABASE_POINTOUT 6-47
*EM_DATABASE_ROGO 6-48
*EM_DATABASE_TIMESTEP 6-49
*EM_EP_CELLMODEL_DEFINE_FUNCTION 6-50
*EM_EP_CELLMODEL_FENTONKARMA 6-53
*EM_EP_CELLMODEL_FITZHUGHNAGUMO 6-56
*EM_EP_CELLMODEL_TENTUSSCHER 6-58
*EM_EP_CELLMODEL_TOMEK 6-67
*EM_EP_CELLMODEL_TOR_ORD 6-68
*EM_EP_CELLMODEL_USERMAT 6-69

TABLE OF CONTENTS

*EM_EP_CREATEFIBERORIENTATION....6-70
*EM_EP_ECG....6-72
*EM_EP_ISOCH....6-73
*EM_EP_LAPLACE_DIRICHLET....6-74
*EM_EP_PURKINJE_NETWORK....6-75
*EM_EP_TENTUSSCHER_STIMULUS....6-77
*EM_EOS_BURGESS....6-78
*EM_EOS_MEADON....6-82
*EM_EOS_PERMEABILITY....6-85
*EM_EOS_TABULATED1....6-86
*EM_EOS_TABULATED2....6-87
*EM_EXTERNAL_FIELD....6-89
*EM_ISOPOTENTIAL....6-90
*EM_ISOPOTENTIAL_CONNECT....6-91
*EM_ISOPOTENTIAL_ROGO....6-93
*EM_MAT_001....6-94
*EM_MAT_002....6-96
*EM_MAT_003....6-98
*EM_MAT_004....6-102
*EM_MAT_005....6-104
*EM_MAT_006....6-108
*EM_OUTPUT....6-110
*EM_OUTPUT_VTK....6-112
*EM_PERMANENT_MAGNET....6-113
*EM_POINT_SET....6-118
*EM_RANDLES_BATMAC....6-120
*EM_RANDLES_EXOTHERMIC_REACTION....6-128
*EM_RANDLES_MESHLESS....6-131
*EM_RANDLES_TSHELL....6-137
*EM_RANDLES_SHORT....6-144
*EM_RANDLES_SOLID....6-148
*EM_ROTATION_AXIS....6-155
*EM_SOLVER_BEM....6-156
*EM_SOLVER_BEMMAT....6-158
*EM_SOLVER_FEM....6-159
*EM_SOLVER_FEMBEM....6-161
*EM_SOLVER_FEMBEM_MONOLITHIC....6-162

*ICFD 7-1
*ICFD_BOUNDARY_CONJ_HEAT 7-5
*ICFD_BOUNDARY_CONVECTION_TEMP 7-6
*ICFD_BOUNDARY_FLUX_TEMP 7-7
*ICFD_BOUNDARY_FREESLIP 7-8
*ICFD_BOUNDARY_FSI 7-9
*ICFD_BOUNDARY_FSI_EXCLUDE 7-10
*ICFD_BOUNDARY_FSI_FIXED 7-11
*ICFD_BOUNDARY_FSWAVE 7-12
*ICFD_BOUNDARY_GROUND 7-15
*ICFD_BOUNDARY_NAVIERSLIP 7-16
*ICFD_BOUNDARY_NONSLIP 7-17
*ICFD_BOUNDARY_PERIODIC 7-18
*ICFD_BOUNDARY_PRESCRIBED_LEVELSET 7-20
*ICFD_BOUNDARY_PRESCRIBED_MOVEMESH 7-21
*ICFD_BOUNDARY_PRESCRIBED_PRE 7-22
*ICFD_BOUNDARY_PRESCRIBED_SPTRANSP_CONC 7-23
*ICFD_BOUNDARY_PRESCRIBED_TEMP 7-24
*ICFD_BOUNDARY_PRESCRIBED_TURBULENCE 7-25
*ICFD_BOUNDARY_PRESCRIBED_VEL 7-28
*ICFD_BOUNDARY_WEAKVEL 7-30
*ICFD_BOUNDARY_WINDKESSEL 7-31
*ICFD_CONTROL_ADAPT 7-33
*ICFD_CONTROL_ADAPT_SIZE 7-35
*ICFD_CONTROL_BACKFLOW 7-36
*ICFD_CONTROL_CONJ 7-37
*ICFD_CONTROL_DEM_COUPLING 7-38
*ICFD_CONTROL_EMBEDSHELL 7-40
*ICFD_CONTROL_FSI 7-41
*ICFD_CONTROL_GAP 7-44
*ICFD_CONTROL_GENERAL 7-45
*ICFD_CONTROL_IMPOSED_MOVE 7-47
*ICFD_CONTROL_LEVELSET 7-51
*ICFD_CONTROL_LOAD 7-52
*ICFD_CONTROL_MESH 7-53
*ICFD_CONTROL_MESH_MOV 7-55

TABLE OF CONTENTS

*ICFD_CONTROL_MONOLITHIC 7-56
*ICFD_CONTROL_OUTPUT 7-57
*ICFD_CONTROL_OUTPUT_SUBDOM 7-60
*ICFD_CONTROL_OUTPUT_VAR 7-62
*ICFD_CONTROL_PARTITION 7-64
*ICFD_CONTROL_POROUS 7-65
*ICFD_CONTROL_STEADY 7-66
*ICFD_CONTROL_SURFMESH 7-67
*ICFD_CONTROL_TAVERAGE 7-68
*ICFD_CONTROL_TIME 7-69
*ICFD_CONTROL_TRANSIENT 7-71
*ICFD_CONTROL_TURBULENCE 7-72
*ICFD_CONTROL_TURB_SYNTHESIS 7-82
*ICFD_DATABASE_AVERAGE 7-83
ICFD_DATABASE_DRAG_*{OPTION} 7-84
*ICFD_DATABASE_FLUX 7-86
*ICFD_DATABASE_FLUX_SURF 7-87
*ICFD_DATABASE_FORCE_DEM 7-88
*ICFD_DATABASE_HTC 7-89
*ICFD_DATABASE_NODEAVG 7-91
*ICFD_DATABASE_NODOUT 7-92
*ICFD_DATABASE_NTEMPOUT 7-93
*ICFD_DATABASE_POINTAVG 7-94
*ICFD_DATABASE_POINTOUT 7-95
*ICFD_DATABASE_RESIDUALS 7-97
*ICFD_DATABASE_SSOUT 7-98
*ICFD_DATABASE_SSOUT_EXCLUDE 7-99
*ICFD_DATABASE_TEMP 7-100
*ICFD_DATABASE_TIMESTEP 7-101
*ICFD_DATABASE_UINDEX 7-102
*ICFD_DATABASE_WETNESS 7-103
*ICFD_DEFINE_HEATSOURCE 7-104
*ICFD_DEFINE_RESIDENCETIMESOURCE 7-105
*ICFD_DEFINE_SOURCE 7-106
*ICFD_DEFINE_SPTRANSPSOURCE 7-107
*ICFD_DEFINE_TURBSOURCE 7-108
*ICFD_DEFINE_POINT 7-110

*ICFD_DEFINE_NONINERTIAL 7-112
*ICFD_DEFINE_WAVE_DAMPING 7-115
*ICFD_INITIAL 7-117
*ICFD_INITIAL_LEVELSET 7-118
*ICFD_INITIAL_SPTRANSP 7-120
*ICFD_INITIAL_TEMPNODE 7-121
*ICFD_INITIAL_TURBULENCE 7-122
*ICFD_MAT 7-123
*ICFD_MODEL_NONNEWT 7-126
*ICFD_MODEL_POROUS 7-130
*ICFD_MODEL_SPECIES_TRANSPORT 7-139
*ICFD_PART 7-140
*ICFD_PART_VOL 7-141
*ICFD_SECTION 7-143
*ICFD_SET_NODE_LIST 7-144
*ICFD_SOLVER_SPLIT 7-145
*ICFD_SOLVER_TOL_FSI 7-146
*ICFD_SOLVER_TOL_LSET 7-147
*ICFD_SOLVER_TOL_MMOV 7-148
*ICFD_SOLVER_TOL_MOM 7-149
*ICFD_SOLVER_TOL_MONOLITHIC 7-150
*ICFD_SOLVER_TOL_PRE 7-151
*ICFD_SOLVER_TOL_TEMP 7-152

*MESH 8-1
*MESH_BL 8-2
*MESH_BL_SYM 8-6
*MESH_EMBEDSHELL 8-7
*MESH_INTERF 8-8
*MESH_NODE 8-9
*MESH_SIZE 8-10
*MESH_SIZE_SHAPE 8-11
*MESH_SURFACE_ELEMENT 8-15
*MESH_SURFACE_NODE 8-17
*MESH_VOLUME 8-18
*MESH_VOLUME_ELEMENT 8-19
*MESH_VOLUME_NODE 8-20

TABLE OF CONTENTS

*MESH_VOLUME_PART 8-21

*STOCHASTIC 9-1

*STOCHASTIC_SPRAY_PARTICLES 9-2

*STOCHASTIC_TBX_PARTICLES 9-7

*LSO 10-1

*LSO_DOMAIN 10-2

*LSO_ID_SET 10-5

*LSO_POINT_SET 10-7

*LSO_TIME_SEQUENCE 10-9

*LSO_VARIABLE_GROUP 10-12

LS-DYNA MULTIPHYSICS USER'S MANUAL

INTRODUCTION

In this manual, there are five main solvers: two compressible flow solvers, an incompressible flow solver, an electromagnetism solver, and a battery electrochemistry solver. Each of them implements coupling with the structural solver in LS-DYNA.

The keywords covered in this manual fit into one of three categories. In the first category are the keyword cards that provide input to each of the multiphysics solvers that in turn couple with the structural solver. In the second category are keyword cards involving extensions to the basic solvers. Presently, the chemistry and stochastic particle solvers are the two solvers in this category, and they are used in conjunction with the *CESE compressible flow solver discussed below. In the third category are keyword cards for support facilities. A volume mesher that creates volume tetrahedral element meshes from bounding surface meshes is one of these tools. Another is a data output mechanism for a limited set of variables from some of the solvers in this manual. This mechanism is accessed through *LSO keyword cards.

The CESE solver is a compressible flow solver based upon the Conservation Element/Solution Element (CE/SE) method, originally proposed by Chang of the NASA Glenn Research Center. This method is a novel numerical framework for conservation laws. It has many non-traditional features, including a unified treatment of space and time, the introduction of separate conservation elements (CE) and solution elements (SE), and a novel shock capturing strategy without using a Riemann solver. This method has been used to solve many types of flow problems, such as detonation waves, shock/acoustic wave interaction, cavitating flows, supersonic liquid jets, and chemically reacting flows. In LS-DYNA, it has been extended to also solve fluid-structure interaction (FSI) problems. It does this with two approaches. The first approach solves the compressible flow equations on an Eulerian mesh while the structural mechanics is solved on a moving mesh that moves through the fixed CE/SE mesh. In the second approach (new with this version), the CE/SE mesh moves in a fashion such that its FSI boundary surface matches the corresponding FSI boundary surface of the moving structural mechanics mesh. This second approach is more accurate for FSI problems, especially with boundary layers flows. Another feature with the CESE moving mesh solver is conjugate heat transfer coupling with the solid thermal solver. The chemistry and stochastic particle solvers are two add-on solvers that extend the CESE solver.

The dual CESE solver is another compressible flow solver that is also based upon the Conservation Element/Solution Element (CE/SE) method, but with improvements related to accuracy and robustness. This method follows a similar novel numerical

framework for conservation laws. In LS-DYNA, the dual CESE solver also include fluid-structure interaction (FSI) capabilities. It also does this with two approaches. The first approach solves the compressible flow equations on an Eulerian mesh while the structural mechanics is solved on a moving mesh that moves through the fixed dual CE/SE mesh. In the second approach. the dual CE/SE mesh moves in a fashion such that its FSI boundary surface matches the corresponding FSI boundary surface of the moving structural mechanics mesh. This second approach is more accurate for FSI problems, especially with boundary layers flows. One of the advances in these FSI computations with the dual CESE solver as compared with the older CESE solver is that each FSI approach (or fixed mesh Eulerian solver) may be employed in different subregions of the fluid mesh in the same problem. Unlike the *CESE solvers, the dual CESE solvers do not yet have conjugate heat transfer coupling with the solid thermal solver, nor coupling with the chemistry or stochastic particle solvers. Another advance available only with the dual CESE solvers is the availability of equations of state for pure and pseudo-pure fluids of industrial interest in the REFPROP and COOLPROP EOS libraries. These complex EOSes are generally expensive to evaluate, so a bi-cubic table look-up mechanism has been developed that greatly accelerates their use.

The third solver is the incompressible flow solver (ICFD) that is fully coupled with the solid mechanics solver. This coupling permits robust FSI analysis via either an explicit technique when the FSI is weak, or using an implicit coupling when the FSI coupling is strong. In addition to being able to handle free surface flows, there is also a bi-phasic flow capability that involves modeling using a conservative Lagrangian interface tracking technique. Basic turbulence models are also supported. This solver is the first in LS-DYNA to make use of a new volume mesher that takes surface meshes bounding the fluid domain as input (*MESH keywords). In addition, during the time advancement of the incompressible flow, the solution is adaptively re-meshed as an automatic feature of the solver. Another important feature of the mesher is the ability to create boundary layer meshes. These anisotropic meshes become a crucial part of the model when shear stresses are to be calculated near fluid walls. The ICFD solver is also coupled to the solid thermal solver using a monolithic approach for conjugate heat transfer problems.

The fourth solver is an electromagnetics (EM) solver. This module solves the Maxwell equations in the Eddy current (induction-diffusion) approximation. This is suitable for cases where the propagation of electromagnetic waves in air (or vacuum) can be considered as instantaneous. Therefore, the wave propagation is not solved. The main applications are Magnetic Metal Forming, bending or welding, induced heating, ring expansions and so forth. The EM module allows the introduction of a source of electrical current into solid conductors and the computation of the associated magnetic field, electric field, as well as induced currents. The EM solver is coupled with the structural mechanics solver (the Lorentz forces are added to the mechanics equations of motion), and with the structural thermal solver (the ohmic heating is added to the thermal solver as an extra source of heat). The EM fields are solved using a Finite Element Method (FEM) for the conductors and a Boundary Element Method (BEM) for the surrounding air/insulators. Thus no air mesh is necessary.

The fifth solver is a battery electrochemistry solver. At this time, the available capability involves a one-dimensional electrochemistry solver that is coupled to the structural mechanics and structural thermal solver in each structural element identified as being part of a battery cell. It solves these one-dimensional models implicitly so as to be able to run simulations for very long physical times that are typical of battery-structure interaction problems.

As stated above, the *CHEMISTRY and *STOCHASTIC cards are only used in the CESE solver at this time.

*BATTERY

The keyword *BATTERY provides input data for the electrochemistry solver:

*BATTERY_DATABASE_HISTORY_ALLNDS_ON_ELESET

*BATTERY_DATABASE_HISTORY_GLOBALS

*BATTERY_DATABASE_HISTORY_NODELIST_ON_ELESET

*BATTERY_ECHEM_CELL_GEOMETRY

*BATTERY_ECHEM_CONTROL_SOLVER

*BATTERY_ECHEM_INITIAL

*BATTERY_ECHEM_MAT_ANODE

*BATTERY_ECHEM_MAT_CATHODE

*BATTERY_ECHEM_MAT_ELECTROLYTE

*BATTERY_ECHEM_THERMAL

For now, the available capability involves a one-dimensional electrochemistry solver that is coupled to the structural mechanics and structural thermal solver in each structural element identified as being part of a battery cell.

*BATTERY_DATABASE_HISTORY_ALLNDS_ON_ELESET

Purpose: Enable output of battery electrochemistry solver data for all one-dimensional battery mesh nodes on each element in an element set. The output goes to binary database binout.

Card 1	1	2	3	4	5	6	7	8
Variable	DT	LCUR	IOOPT	ESID				
Type	F	I	I	I				
Default	0.0	0	0	none				

VARIABLE	DESCRIPTION
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_n)$. This is the default behavior. EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1})$. EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.
ESID	ID of a *SET_SOLID element set to use for the binout operation. The selected elements in the element set of volume structural elements must also be battery electrochemistry elements.

*BATTERY_DATABASE_HISTORY_NODELIST_ON_ELESET

Purpose: Enable output of battery electrochemistry solver data for a specified set of one-dimensional battery mesh nodes on each element in an element set. The output goes to binary database binout.

Card 1	1	2	3	4	5	6	7	8
Variable	DT	LCUR	IOOPT	ESID				
Type	F	I	I	I				
Default	0.0	0	0	none				

Node ID Cards. Battery mesh nodes to select for battery data output in the elements selected by ESID in Card 1. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	NID1	NID2	NID3	NID4	NID5	NID6	NID7	NID8
Type	I	I	I	I	I	I	I	I

VARIABLE	DESCRIPTION
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $$t_{n+1} = t_n + \mathrm{LCUR}(t_n)\ .$$ This is the default behavior.

VARIABLE	DESCRIPTION
	EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $$t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1}) \ .$$
	EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.
ESID	ID of a *SET_SOLID element set to use for the binout operation. The selected elements in the element set of volume structural elements must also be battery electrochemistry elements.
NID*i*	Node ID *i*

*BATTERY_DATABASE_HISTORY_GLOBALS

Purpose: Enable output of battery electrochemistry solver data. The output goes to binary database binout.

Card 1	1	2	3	4	5	6	7	8
Variable	DT	LCUR	IOOPT					
Type	F	I	I					
Default	0.0	0	0					

VARIABLE	DESCRIPTION
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_n)$. This is the default behavior. EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1})$. EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.

*BATTERY_ECHEM_CELL_GEOMETRY

Purpose: Set general-purpose geometry variables for a single cell BATTERY model.

Card 1	1	2	3	4	5	6	7	8
Variable	IMODEL	ALEN	SLEN	CLEN	ACCLEN	CCCLEN		
Type	I	F	F	F	F	F		
Default	none	none	none	none	none	none		

Card 2	1	2	3	4	5	6	7	8
Variable	NELEA	NELES	NELEC	NELECCA	NELECCC			
Type	I	I	I	I	I			
Default	none	none	none	none	none			

VARIABLE	DESCRIPTION
IMODEL	Battery model identifier (see Remark 1)
ALEN	Length of the anode side electrode
SLEN	Length of the separator
CLEN	Length of the cathode side electrode
ACCLEN	Length of the negative current collector
CCCLEN	Length of the positive current collector
NELEA	Number of elements in the anode electrode
NELES	Number of elements in the separator
NELEC	Number of elements in the cathode electrode
NELECCA	Number of elements in the anode current collector

VARIABLE	DESCRIPTION
NELECCC	Number of elements in the cathode current collector

Remarks:

1. **Identifier.** The battery model identifier (IMODEL) should match the IMODEL value specified in the corresponding *BATTERY_ECHEM_CONTROL_-SOLVER card. If a different value is given, the value on the *BATTERY_-ECHEM_CONTROL_SOLVER card will be the default.

*BATTERY_ECHEM_CONTROL_SOLVER

Purpose: Set general purpose control variables for a battery electrochemistry simulation.

Card Summary:

Card 1. This card is required.

IMODEL	IDIMEN	NCYCLE	IAGING	ITRA	IGAS		

Card 2. Include NCYCLE instances of this card, one for each cycle.

CMODE	CTYPE	CEND	TCUT	VCUT	RCURR		

Card 3.1. Include this card if IAGING = 1.

SEIMW	SEIRHO	SEIBRUG	SEIEPS	SEICO	SEITO		

Card 3.2. Include this card if IAGING = 1.

SEII0	SEIRKA	SEICON	ECCO	ECDFS			

Card 4.1. Include this card if ITRA = 1.

AFI	EAT	HOFEC	HOFLI	HOFLED			

Card 4.2. Include this card if ITRA = 1.

HOFC2H4	HOFLC	HOFCO2	HOFO2				

Card 5. Include this card if IGAS = 1.

IC2H4	IO2	ICO2	IH2O	AG1	AG2	EG1	EG2

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	IMODEL	IDIMEN	NCYCLE	IAGING	ITRA	IGAS		
Type	I	I	I	I	I	I		
Default	none	none	1	1	0	0		

VARIABLE	DESCRIPTION
IMODEL	Sets the battery model: EQ.1: Dual insertion model (Newman). IAGING, ITRA, and IGAS must be set to 0. EQ.2: Multiphysics LIB model. With IGAS = 0, the model includes thermal effects while with IGAS = 1, the model includes both thermal and chemistry effects. Note that IAGING and ITRA must be set to 1 for either case. See Remark 1. EQ.3: Lithium metal battery
IDIMEN	Sets the geometric dimension: EQ.1: 1D LIB models EQ.101: 1D models with thermo-mechanical coupling (available for IMODEL = 1 and 2). See Remark 2.
NCYCLE	Number of cycles to run. Default is 1 cycle.
IAGING	Aging model flag (see Remark 3): EQ.0: Off EQ.1: On
ITRA	Thermal runaway model flag (see Remark 3): EQ.0: Off EQ.1: On
IGAS	Gas generation model flag (see Remark 4): EQ.0: Off EQ.1: On

Cycle Card. Include NCYCLE instances of this card, that is, one for each cycle.

Card 2	1	2	3	4	5	6	7	8
Variable	CMODE	CTYPE	CEND	TCUT	VCUT	RCURR		
Type	I	I	I	F	F	F		
Default	1	0	none	none	none	none		

VARIABLE	DESCRIPTION
CMODE	Battery running mode flag: EQ.0: Galvanostatic run EQ.1: Potentiostatic run (This is under development and not recommended.)
CTYPE	Running current type: EQ.0: Constant current EQ.1: Variable current
CEND	Cause of battery cycle termination: EQ.1: Cycle run for a given time period. EQ.2: Cycle run until a given cut-off voltage.
TCUT	Total running time for the cycle
VCUT	Cut-off voltage for the cycle
RCURR	Cycle operating current in the case of constant current

Aging Card 1. Include this card if IAGING = 1.

Card 3.1	1	2	3	4	5	6	7	8
Variable	SEIMW	SEIRH0	SEIBRUG	SEIEPS	SEIC0	SEIT0		
Type	F	F	F	F	F	F		
Default	0.0	0.0	0.0	0.0	0.0	0.0		

Aging Card 2. Include this card if IAGING = 1.

Card 3.2	1	2	3	4	5	6	7	8
Variable	SEII0	SEIRKA	SEICON	ECC0	ECDFS			
Type	F	F	F	F	F			
Default	0.0	0.0	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
SEIMW	Molecular weight of the SEI
SEIRHO	Density of the SEI
SEIBRUG	The Bruggeman constant of the SEI
SEIEPS	Initial SEI porosity
SEIC0	Initial SEI concentration (units: mol/m^3)
SEIT0	Initial thickness of the SEI layer (units: m)
SEII0	Exchange current density for the SEI reaction
SEIRKA	Reaction rate constant for the SEI reaction (ignored if SEII0 ≠ 0.0)
SEICON	Ionic conductivity (units: S/m)
ECC0	Initial concentration of EC (ethylene carbonate). This field is ignored if SEII0 ≠ 0.0.

VARIABLE	DESCRIPTION
ECDFS	Diffusion coefficient of EC

Thermal Runaway Card 1. Include this card if ITRA = 1.

Card 4.1	1	2	3	4	5	6	7	8
Variable	AFI	EAT	HOFEC	HOFLI	HOFLED			
Type	F	F	F	F	F			
Default	0.0	0.0	0.0	0.0	0.0			

Thermal Runaway Card 2. Include this card if ITRA = 1.

Card 4.2	1	2	3	4	5	6	7	8
Variable	HOFC2H4	HOFLC	HOFCO2	HOFO2				
Type	F	F	F	F				
Default	0.0	0.0	0.0	0.0				

VARIABLE	DESCRIPTION
AFI	Frequency factor for the reaction
EAT	Activation energy for the reaction
HOFEC	Formation enthalpy of the EC (units: kJ/mol)
HOFLI	Formation enthalpy of the LI
HOFLED	Formation enthalpy of the SEI layer (units: kJ/mol)
HOFC2H4	Formation enthalpy of ethylene (units: kJ/mol)
HOFLC	Formation enthalpy of LC (Li_2CO_3; units: kJ/mol)
HOFCO2	Formation enthalpy of CO_2 (units: kJ/mol)
HOFO2	Formation enthalpy of O_2 (units: kJ/mol)

Gas Initial Conditions Card. Include this card if IGAS = 1.

Card 5	1	2	3	4	5	6	7	8
Variable	IC2H4	IO2	ICO2	IH2O	AG1	AG2	EG1	EG2
Type	F	F	F	F	F	F	F	F
Default	0.0	0.0	0.0	0.0	3.426E26	5.028E-6	2.50e5	2.51e5
Remarks					5	5	5	5

VARIABLE	DESCRIPTION
IC2H4	Initial concentration of C_2H_2 gas (units: mol/m^3)
IO2	Initial concentration of O_2 gas (units: mol/m^3)
ICO2	Initial concentration of CO_2 gas (units: mol/m^3)
IH2O	Initial concentration of H_2O gas (units: mol/m^3)
AG1	Frequency factor for Ethylene oxidation (units: m/s)
AG2	Frequency factor for Lithium hydration (units: m/s)
EG1	Activation energy of Ethylene oxidation (units: J/mol)
EG2	Activation energy of Lithium hydration (units: J/mol)

Remarks:

1. **Multiphysics Battery Models.** With IMODEL = 2, you can model a battery with thermal effects or a battery with both thermal and chemistry effects depending on the value of IGAS. These two battery models include battery aging, thermal runaway, and battery swelling. The battery and chemistry effects model additionally includes a gas generation model for modeling the chemistry effects.

2. **Shutdown Key.** For coupled battery models (IDIMEN = 101) with IMODEL = 1 or 2, the shutdown key will work if the minimum ignition energy in the battery system reaches the critical condition like pre-thermal-runaway.

3. **Input Requirements for Aging and Thermal Runaway Models.** When IAGING = 1, all the variables on Cards 3.1 and 3.2 must be filled. A similar

requirement applies when ITRA = 1 in that all the variables of Cards 41 and 4.2 must be filled. For more details about these models, please see the Theory manual.

4. **Battery Chemistry Files.** When IGAS = 1, *CHEMISTRY_BATTERY must be included in the input deck to provide the file names for the battery chemistry input file, the corresponding thermodynamics data file, and the transport properties file. Please refer to the *CHEMISTRY chapter for further details about the *CHEMISTRY_BATTERY keyword.

5. **AG1, AG2, EG1, and EG2.** AG1, AG2, EG1, and EG2 are used to compute the rate of reaction based on Arrhenius equation as

$$r = A_k c_k e^{-\frac{\mathrm{EG}_k}{RT}} \quad .$$

A_k is the frequency factor for the reaction, c_k is the concentration, EG_k is the activation energy for the reaction, R is the universal gas constant, and T is the temperature.

*BATTERY_ECHEM_INITIAL

Purpose: Initializes the composite electrodes and electrolyte in every element of the BATTERY simulation mesh.

Card 1	1	2	3	4	5	6	7	8
Variable	DT0	LICE	PHI1	LICS	CURRIC	FLUXIC	PHI2	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
DT0	Initial time step size
LICE	Initial concentration of lithium ions in the electrolyte
PHI1	Initial condition of the electrolyte potential
LICS	Initial concentration of lithium in the solid particles
CURRIC	Initial current of the cycle operation
FLUXIC	Initial pore wall flux
PHI2	Initial condition of the electrode potential

*BATTERY_ECHEM_MAT_ANODE

Purpose: Set the battery material variables for the anode side electrode.

Card 1	1	2	3	4	5	6	7	8
Variable	PIDA	IOCPA	CAPTA	SOCA	RADA	RATEA	RANODE	
Type	I	I	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

Card 2	1	2	3	4	5	6	7	8
Variable	RHOEA	RHOFA	RHOCCA	DFSA	CONDA	MWA		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Card 3	1	2	3	4	5	6	7	8
Variable	VFEA	VFPA	VFFA	VFGA				
Type	F	F	F	F				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
PIDA	Part ID
IOCPA	Material type for the OCP in the anode electrode: EQ.1: Sony carbon (petroleum coke) EQ.2: MCMB 2510 EQ.3: MCMB 2528

VARIABLE	DESCRIPTION
	EQ.4: KS6 graphite
CAPTA	Coulombic capacity of anode material (units: mAh/g)
SOCA	Initial lithium stoichiometric coefficient of the anode side active material. For example, Li_xC_6 ($0 < x < 0.7$).
RADA	Radius of spherical particles in the anode side active material (units: m)
RATEA	Reaction rate constant for the anode electrode
RANODE	Film resistance for the anode electrode
RHOEA	Density of anode insertion material (electrode particles) (kg/m^3)
RHOFA	Density of the inert filler in the anode (units: kg/m^3)
RHOCCA	Density of the current collector in the anode (units: kg/m^3)
DFSA	Diffusion coefficient of lithium ions in the anode electrode material (units: m^2/s)
CONDA	Effective electronic conductivity of the anode porous electrode (units: S/m)
MWA	Molecular weight of the anode electrode (units: kg/mol)
VFEA	Volume fraction of electrolyte in the anode electrode
VFPA	Volume fraction of the polymer phase in the anode electrode
VFFA	Volume fraction of the inert filler in the anode electrode
VFGA	Volume fraction of the gas in the anode electrode

*BATTERY_ECHEM_MAT_CATHODE

Purpose: Set the battery material variables for the positive electrode.

Card 1	1	2	3	4	5	6	7	8
Variable	PIDC	IOCPC	CAPTC	SOCC	RADC	RATEC	RCATHDE	
Type	I	I	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

Card 2	1	2	3	4	5	6	7	8
Variable	RHOEC	RHOFC	RHOCCC	DFSC	CONDC	MWC		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Card 3	1	2	3	4	5	6	7	8
Variable	VFEC	VFPC	VFFC	VFGC				
Type	F	F	F	F				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
PIDC	Part ID
IOCPC	Material type for the open-circuit potential: EQ.1: Mn_2O_4(lower plateau) (1.1 < y < 1.99) EQ.2: Mn_2O_4 (upper plateau) (0.17 < y < 0.99) EQ.3: Cobalt dioxide 1, Li_yCoO_2 (0.0 < y < 0.99)

VARIABLE	DESCRIPTION
	EQ.4: Cobalt dioxide 2, Li_yCoO_2 ($0.0 < y < 0.99$)
	EQ.5: Mn_2O_4 (literature version) ($0.17 < y < 0.99$)
	EQ.6: NMC-111
	EQ.7: NMC-811
CAPTC	Coulombic capacity of the cathode material (units: mAh/g)
SOCC	Initial lithium stoichiometric coefficient for the cathode side active material. For example, Li_yWO_3 ($0 < y < 0.67$).
RADC	Radius of spherical particle in the cathode side active material. (units: m)
RATEC	Reaction rate constant for the cathode electrode
RCATH	Film resistance for the cathode electrode
RHOEC	Density of the cathode insertion material (electrode particles). (units: kg/m^3)
RHOFC	Density of the cathode side inert filler (units: kg/m^3)
RHOCCC	Density of the cathode side current collector (units: kg/m^3)
DFSC	Diffusion coefficient of lithium ions in the cathode insertion material (units: m^2/s)
CONDC	Effective electronic conductivity of the cathode porous electrode (units: S/m).
MWC	Molecular weight of the cathode electrode (units: kg/mol)
VFEC	Volume fraction of electrolyte in the cathode electrode
VFPC	Volume fraction of the polymer phase in the cathode electrode
VFFC	Volume fraction of the inert filler in the cathode electrode
VFGC	Volume fraction of the gas in the cathode electrode

*BATTERY_ECHEM_MAT_ELECTROLYTE

Purpose: Set the battery material variables for the electrolyte and separator.

Card 1	1	2	3	4	5	6	7	8
Variable	PIDEL	IOCPE	IELTYPE	RHOEL	RHOP	RHOS		
Type	I	I	I	F	F	F		
Default	none	none	none	none	none	none		

Card 2	1	2	3	4	5	6	7	8
Variable	VFES	VFPS	VFGS					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
PIDEL	Part ID
IOPCE	Material type for the open-circuit potential: EQ.1: LiPF6 in EC : DMC (1:1). EQ.2: LiPF6 in EC : DMC (2:1). EQ.3: LiPF6 in EC : DMC (1:2). EQ.4: LiPF6 in PC EQ.5: LiClO4 in PC Here, EC is ethylene carbonate, DMC is dimethyl carbonate, and PC is propylene carbonate.
IELTYPE	Type of electrolyte (units: kg/m^3): EQ.0: Liquid electrolyte EQ.1: Polymer electrolyte

VARIABLE	DESCRIPTION
RHOEL	Density of the electrolyte (units: kg/m^3)
RHOP	Density of the polymer phase (units: kg/m^3)
RHOS	Density of the separator material (units: kg/m^3)
VFES	Volume fraction of electrolyte in the separator
VFPS	Volume fraction of the polymer phase in the separator
VFGS	Volume fraction of the gas in the separator

*BATTERY_ECHEM_THERMAL

Purpose: Set parameters for the thermal treatment in a cell stack.

Card 1	1	2	3	4	5	6	7	8
Variable	TNAME	ITTYPE	IPRT	CP	HCONV	TEMP		
Type	A	I	I	F	F	F		
Default	none	none	none	none	none	none		
Remarks			3	2	2	2		

Card 2	1	2	3	4	5	6	7	8
Variable	FILE							
Type	A							

VARIABLE	DESCRIPTION
TNAME	Thermal material identifier
ITTYPE	Flag for how temperature is determined: EQ.0: Constant temperature mode EQ.1: Isothermal temperature with time EQ.2: Thermal coupling with LS-DYNA thermal solver
IPRT	Data print in ASCII format: EQ.0: No data print out. EQ.1: Time versus heat flux print out for thermal solver
CP	Specific heat coefficient of the cell (units: J/(kg K))
HCONV	Convective heat transfer coefficient with external medium. (units: $W/(m^2K)$)
TEMP	Ambient temperature around the cell stack (K)

VARIABLE	DESCRIPTION
FILE	Name of the battery cell output file (ASCII)

Remarks:

1. **Battery Parts for Thermal-Mechanical Coupling.** In case of thermal-mechanical coupling, the part number for the battery simulation must be specified, so only this part number is considered in the battery parts.

2. **Material Properties for Thermal-Mechanical Coupling.** If ITTYPE is 2, the material properties are set through the thermal material card, including anisotropic conductivities (see *MAT_THERMAL_ORTHOTROPIC). CP, HCONV, and TEMP specified here are ignored.

3. **Heat Flux Output.** If IPRT = 1, then the heat flux generated by the battery solver is printed out.

Example:

The following is a partial example for 1D Electrochemisty.

```
*Keyword
$
*TITLE
1D battery models
$
*BATTERY_ECHEM_CONTROL_SOLVER
$--------1---------2---------3---------4---------5---------6---------7
$   imodel    idimen    ncycle    iaging      itra      igas
         2         1         1         1         1         1
$    cmode     ctype      cend      tcut      vcut     rcurr
         0         0         2       0.0       3.0      8.75
$ aging cards
$    seimw    seirho   seibrug    seieps     seic0     seit0
       0.0       0.0       0.0       0.0       0.0       0.0
$    seii0    seirka    seicon      ecc0     ecdfs
       0.0       0.0       0.0       0.0       0.0
$ thermal runaway cards
$      afi       eat     hofec     hofli    hofled
       0.0       0.0       0.0       0.0       0.0
$  hofc2h4     hoflc    hofco2     hofo2
       0.0       0.0       0.0       0.0
$ gas initial conditions card
     ic2h4       io2       ico2      ih2o       ag1       ag2       eg1       eg2
   1.0e-10   1.0e-10   1.0e-10   1.0e-10       0.0       0.0       0.0       0.0
*CHEMISTRY_BATTERY
  battery.inp
  btherm.dat
  btran.dat
$
*BATTERY_ECHEM_INITIAL
$--------1---------2---------3---------4---------5---------6---------7
$      dt0      lice      phi1      lics    curric    fluxic      phi2
      0.02    1000.0       0.0      0.05       0.0       5.0   -1.0e-7
$
```

```
*BATTERY_ECHEM_CELL_GEOMETRY
$--------1---------2---------3---------4---------5---------6---------7
$   imodel      alen      slen      clen     acclen    ccclen
         2    9.6e-5    2.5e-5   6.0e-5     1.0e-5    1.0e-5
$    nelea     neles     nelec
        40        40        80
$
*BATTERY_ECHEM_MAT_ANODE
$--------1---------2---------3---------4---------5---------6---------7
$     pida     i0cpa      capta      soca      rada     ratea    ranode
         2         3     372.0       0.6    8.0e-6    3.0e-9   0.35e-2
$    rhoea     rhofa    rhocca      dfsa     conda       mwa
    1800.0    1800.0    8954.0   7.0e-14     100.0     0.079
$     vfec      vfpc      vffc      vfgc
       0.4       0.0     0.064       0.0
$
*BATTERY_ECHEM_MAT_CATHODE
$--------1---------2---------3---------4---------5---------6---------7
$     pidc     iocpc      captc      socc      radc     ratec   rcathde
         2         3     274.0       0.8    5.0e-6    3.0e-9       0.0
$    rhoec     rhofc    rhoccc      dfsc     condc       mwc
    5010.0    1800.0    2707.0   3.0e-14       0.5    0.9787
$    vfelc     vfplc     vffic     vfgsx
      0.36       0.0     0.106       0.0
$
*BATTERY_ECHEM_MAT_ELECTROLYTE
$--------1---------2---------3---------4---------5---------6---------7
$    pidel     iocpe   ieltype     rhoel      rhop      rhos
         2         1         0    1324.0    1780.0     552.0
$    vfels     vfpls     vfgss
       0.4       0.0       0.0
$
*BATTERY_ECHEM_THERMAL
$--------1---------2---------3---------4---------5---------6---------7
$    tname    ittype      iprt        cp     hconv      temp
  hot_batt         1         0     500.0       0.0    298.15
heat_discharg_lco.k
$
*END
```

*CESE

The keyword *CESE provides input data for the Conservation Element/Solution Element (CESE) compressible fluid solver:

CESE_BOUNDARY_AXISYMMETRIC_*{OPTION}

CESE_BOUNDARY_BLAST_LOAD}*

CESE_BOUNDARY_CONJ_HEAT_*{OPTION}

CESE_BOUNDARY_CYCLIC_*{OPTION}

CESE_BOUNDARY_FSI_*{OPTION}

CESE_BOUNDARY_NON_REFLECTIVE_*{OPTION}

CESE_BOUNDARY_PRESCRIBED_*{OPTION}

CESE_BOUNDARY_REFLECTIVE_*{OPTION}

CESE_BOUNDARY_SLIDING_*{OPTION}

CESE_BOUNDARY_SOLID_WALL_*{OPTION1}_{OPTION2}

*CESE_CHEMISTRY_D3PLOT

*CESE_CONTROL_LIMITER

*CESE_CONTROL_MESH_MOV

*CESE_CONTROL_SOLVER

*CESE_CONTROL_TIMESTEP

*CESE_DATABASE_ELOUT

*CESE_DATABASE_FLUXAVG

*CESE_DATABASE_FSIDRAG

*CESE_DATABASE_POINTOUT

*CESE_DATABASE_SSETDRAG

*CESE_DEFINE_NONINERTIAL

*CESE_DEFINE_POINT

*CESE_DRAG

*CESE_EOS_CAV_HOMOG_EQUILIB_

*CESE_EOS_IDEAL_GAS

*CESE_EOS_INFLATOR1

*CESE_EOS_INFLATOR2

*CESE_FSI_EXCLUDE

*CESE_INITIAL

CESE_INITIAL_*{OPTION}

*CESE_INITIAL_CHEMISTRY

*CESE_INITIAL_CHEMISTRY_ELEMENT

*CESE_INITIAL_CHEMISTRY_PART

*CESE_INITIAL_CHEMISTRY_SET

*CESE_MAT_000

*CESE_MAT_001 (*CESE_MAT_GAS)

*CESE_MAT_002

*CESE_PART

*CESE_SURFACE_MECHSSID_D3PLOT

*CESE_SURFACE_MECHVARS_D3PLOT

Note that when performing a chemistry calculation with the CESE solver, initialization should only be done with the *CESE_INITIAL_CHEMISTRY_… cards, not the *CESE_-INITIAL… cards.

CESE_BOUNDARY_AXISYMMETRIC_*OPTION

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Define an axisymmetric boundary condition on the axisymmetric axis for the 2D axisymmetric CESE compressible flow solver.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Summary:

Card 1a. This card is included if the keyword option is MSURF. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

MSURFID							

Card 1b. This card is included if the keyword option is MSURF_SET. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

MSURF_S							

Card 1c. This card is included if the keyword option is SET. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

SSID							

Card 1d. This card is included if the keyword option is SEGMENT. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

N1	N2	N3	N4				

Data Card Definitions:

Surface Part Card. Card 1 format used when the MSURF keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.

Set Card. Card 1 format used when the SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
SSID	Segment set ID

Segment Cards. Card 1 format used when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
N1, N2, …	Node IDs defining a segment

Remarks:

1. **Restrictions.** This boundary condition can only be used on the axisymmetric axis for the 2D axisymmetric CESE fluid solver.

CESE_BOUNDARY_BLAST_LOAD_*OPTION

Available options include:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: For the CESE compressible flow solver, set boundary values for velocity, density, and pressure from a blast wave defined by a *LOAD_BLAST_ENHANCED card. Boundary values are applied at the centroid of elements connected with this boundary. *OPTION* = SET and *OPTION* = SEGMENT are for user defined meshes whereas *OPTION* = MSURF and *OPTION* = MSURF_SET are associated with the automatic volume mesher (See *MESH keywords). In other words, the MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Summary:

Card 1a. This card is included if the keyword option is MSURF.

BID	MSURFID						

Card 1b. This card is included if the keyword option is MSURF_SET.

BID	MSURF_S						

Card 1c. This card is included if the keyword option is SET.

BID	SSID						

Card 1d. This card is included if the keyword option is SEGMENT.

BID	N1	N2	N3	N4			

Data Card Definitions:

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	BID	MSURFID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
BID	Blast source ID (see *LOAD_BLAST_ENHANCED)
MSURFID	A mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	BID	MSURF_S						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
BID	Blast source ID (see *LOAD_BLAST_ENHANCED)
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.

Set Card. Card 1 format used when the SET keyword option is active.

Card 1c	1	2	3	4	5	6	7	8
Variable	BID	SSID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
BID	Blast source ID (see *LOAD_BLAST_ENHANCED)
SSID	Segment set ID

Segment Card. Card 1 for SEGMENT keyword option is active.

Card 1d	1	2	3	4	5	6	7	8
Variable	BID	N1	N2	N3	N4			
Type	I	I	I	I	I			
Default	none	none	none	none	none			

VARIABLE	DESCRIPTION
BID	Blast source ID (see *LOAD_BLAST_ENHANCED)
N1, N2, …	Node ID's defining a segment

CESE_BOUNDARY_CONJ_HEAT_*OPTION

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Define a conjugate heat transfer interface condition for CESE compressible flows. This condition identifies those boundary faces of the CESE mesh that are in contact with non-moving structural parts and through which heat flows. This is only possible when the structural thermal solver is also being used in the structural parts.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Summary:

Card 1a. This card is included when the keyword option is MSURF. Include as many cards as necessary. This input ends at the next keyword ("*") card.

MSURFID							

Card 1b. This card is included when the keyword option is MSURF_SET. Include as many cards as necessary. This input ends at the next keyword ("*") card.

MSURF_S							

Card 1c. This card is included when the keyword option is SET. Include as many cards as necessary. This input ends at the next keyword ("*") card.

SSID							

Card 1d. This card is included when the keyword option is SEGMENT. Include as many cards as necessary. This input ends at the next keyword ("*") card.

N1	N2	N3	N4				

Data Card Definitions:

Surface Part Card. Card 1 used when the MSURF keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.

Surface Part Set Card. Card 1 used when the MSURF_SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSURF_S	Identifier of a set of mesh surface part IDs created with an *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.

Set Card. Card 1 used when the SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
SSID	Segment set ID

Segment Cards. Card 1 used when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
N1, N2, …	Node IDs defining a segment

Remarks:

1. **Restrictions.** This boundary condition should only be imposed on a CESE mesh boundary that is in contact with non-moving structural parts. An Eulerian CESE solver is required as is use of the structural thermal solver.

CESE_BOUNDARY_CYCLIC_*OPTION

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Define a cyclic (periodic) boundary condition for CESE compressible flows. This cyclic boundary condition can be used on periodic boundary surfaces.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Summary:

Include as many sets of the following cards as needed. This input ends at the next keyword ("*") card.

Card 1a. This card is included when the keyword option is set to MSURF.

MSURFID1	MSURFID2	CYCTYP					

Card 1b. This card is included when the keyword option is set to MSURF_SET.

MSRF_S1	MSRF_S2	CYCTYP					

Card 1c. This card is included when the keyword option is set to SET.

SSID1	SSID2	CYCTYP					

Card 1d. This card is included when the keyword option is set to SEGMENT.

ND1	ND2	ND3	ND4	NP1	NP2	NP3	NP4

Card 2a. This card is included for the MSURF, MSURF_SET, and SET options when CYCTYP = 1.

AXISX1	AXISY1	AXISZ1	DIRX	DIRY	DIRZ	ROTANG	

Card 2b. This card is included for the MSURF, MSURF_SET, and SET options when CYCTYP = 2.

TRANSX	TRANSY	TRANSZ					

Data Card Definitions:

Surface Part Card. Card 1 format when the MSURF keyword option is active

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID1	MSURFID2	CYCTYP					
Type	I	I	I					
Default	none	none	0					
Remarks	2	2	1					

VARIABLE	DESCRIPTION
MSURFID1, MSURFID2	Mesh surface part IDs referenced in *MESH_SURFACE_ELEMENT cards.
CYCTYP	Relationship between the two cyclic boundary condition surfaces: EQ.0: None assumed (default) EQ.1: The first surface is rotated about an axis to match the second surface. EQ.2: The faces of the first surface are translated in a given direction to obtain the corresponding faces on the second surface.

Surface Part Set Card. Card 1 format when the MSURF_SET keyword option is active

Card 1b	1	2	3	4	5	6	7	8
Variable	MSRF_S1	MSRF_S2	CYCTYP					
Type	I	I	I					
Default	none	none	0					
Remarks	3	3	1					

VARIABLE	DESCRIPTION
MSRF_S1, MSRF_S2	Identifiers of two sets of mesh surface part IDs, each created with a *LSO_ID_SET card, where each mesh surface part ID in each set is referenced in *MESH_SURFACE_ELEMENT cards.
CYCTYP	Relationship between the two cyclic boundary condition surfaces: EQ.0: None assumed (default) EQ.1: The first surface is rotated about an axis to match the second surface. EQ.2: The faces of the first surface are translated in a given direction to obtain the corresponding faces on the second surface.

Set Card. Card 1 format when the SET keyword option is active

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID1	SSID2	CYCTYP					
Type	I	I	I					
Default	none	none	0					
Remarks	4	4	1					

VARIABLE	DESCRIPTION
SSID1, SSID2	A pair of segment set IDs.
CYCTYP	Relationship between the two cyclic boundary condition surfaces: EQ.0: None assumed (default) EQ.1: The first surface is rotated about an axis to match the second surface. EQ.2: The faces of the first surface are translated in a given direction to obtain the corresponding faces on the second surface.

Segment Card. Card 1 format when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	ND1	ND2	ND3	ND4	NP1	NP2	NP3	NP4
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
ND*i*, NP*i*	Node IDs defining a pair of segments: ND1, ND2, ND3, ND4 define the first segment, while NP1, NP2, NP3, NP4 define the second segment. This pair of segments must match either through a geometric translation or rotation.

Rotation Case Card. Additional card for the MSURF, MSURF_SET, and SET options when CYCTYP = 1.

Card 2a	1	2	3	4	5	6	7	8
Variable	AXISX1	AXISY1	AXISZ1	DIRX	DIRY	DIRZ	ROTANG	
Type	F	F	F	F	F	F	F	
Default	0.0	0.0	0.0	none	none	none	none	

VARIABLE	DESCRIPTION
AXIS[Z,Y,Z]1	A point on the axis of rotation
DIR[X,Y,Z]	The direction which together with AXIS[X,Y,Z]1 defines the axis of rotation
ROTANG	The angle of rotation (in degrees) that transforms the centroid of each face on the first surface to the centroid of the corresponding face on the second surface

Translation Case Card. Additional card for the MSURF, MSURF_SET, and SET options when CYCTYP = 2.

Card 2b	1	2	3	4	5	6	7	8
Variable	TRANSX	TRANSY	TRANSZ					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
TRANS[X,Y,Z]	The translation direction that enables the identification of the segment in the second surface that matches a segment in the first surface

Remarks:

1. **CYCTYP.** For the MSURF, MSURF_SET, or SET options with CYCTYP = 0, the code examines the geometry of two faces of the two surfaces in order to determine if the surfaces are approximately parallel (CYCTYP = 2) or related through a rotation (CYCTYP = 1). The geometric parameters required are then computed.

2. **MSURF.** For the MSURF option, the number of mesh surface elements must be the same in each mesh surface part. The mesh surface elements in each mesh surface part are then internally ordered to match pairwise between the two mesh surface parts.

3. **MSURF_SET.** For the MSURF_SET option, the number of mesh surface elements must be the same in each mesh surface part set. The mesh surface elements in each mesh surface part set are then internally ordered to match pairwise between the two mesh surface part sets.

4. **SET.** For the SET option, the same number of segments must be in each set. The segments in each set are then internally ordered to match pairwise between the two sets.

***CESE_BOUNDARY_FSI**_*OPTION*

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Define an FSI boundary condition for the moving mesh CESE compressible flow solver. This card must not be combined with the immersed-boundary method CESE solver, and doing so will result in an error termination condition.

This boundary condition must be applied on a surface of the CESE computational domain that is co-located with surfaces of the outside boundary of the structural mechanics mesh. The nodes of the two meshes will generally not be shared.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Surface Part Card. Card 1 format used when the MSURF keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID							
Type	I							
Default	none							

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S							
Type	I							
Default	none							

Set Card. Card 1 format used when the SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

Segment Cards. Card 1 format used when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.

VARIABLE	DESCRIPTION
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID.
N1, …	Node IDs defining a segment

Remarks:

1. This boundary condition card is also needed for conjugate heat transfer problems with the moving mesh CESE solver.

CESE_BOUNDARY_NON_REFLECTIVE_*OPTION

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Define a passive boundary condition for CESE compressible flows. This non-reflective boundary condition (NBC) provides an artificial computational boundary for an open boundary that is passive.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Surface Part Card. Card 1 used when the MSURF keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID							
Type	I							
Default	none							

Surface Part Set Card. Card 1 used when the MSURF_SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S							
Type	I							
Default	none							

Set Card. Card 1 used when the SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

Segment Cards. Card 1 used when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.
MSURF_S	Identifier of a set of mesh surface part IDs created with an *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID
N1, N2, …	Node IDs defining a segment

Remarks:

1. This boundary condition is usually imposed on an open surface that is far from the main disturbed flow (the further away, the better), i.e., the flow on that boundary surface should be almost uniform.

2. If any boundary segment has not been assigned a boundary condition by any of the *CESE_BOUNDARY_… cards, then it will automatically be assigned this non-reflective boundary condition.

CESE_BOUNDARY_PRESCRIBED_*OPTION

Available options include:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: For the CESE compressible flow solver, set boundary values for velocity, density, pressure and temperature. Boundary values are applied at the centroid of elements connected with this boundary. OPTION = SET and OPTION = SEGMENT are for user defined meshes whereas OPTION = MSURF or MSURF_SET are associated with the automatic volume mesher (See *MESH keywords).

That is, the MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.

2. Card 2 reads in load curve IDs.

3. Card 3 reads in scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID	IDCOMP						
Type	I	I						
Default	none	none						

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S	IDCOMP						
Type	I	I						
Default	none	none						

Set Card. Card 1 format used when the SET keyword option is active.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID	IDCOMP						
Type	I	I						
Default	none	none						

Segment Card. Card 1 for SEGMENT keyword option is active.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N1	N3	N4	IDCOMP			
Type	I	I	I	I	I			
Default	none	none	none	none	none			

Load Curve Card.

Card 2	1	2	3	4	5	6	7	8
Variable	LC_U	LC_V	LC_W	LC_RHO	LC_P	LC_T		
Type	I	I	I	I	I	I		
Remarks	1,2,3	1,2,3	1,2,3	1,2,3	1,2,3	1,2,3		

Scale Factor Card.

Card 3	1	2	3	4	5	6	7	8
Variable	SF_U	SF_V	SF_W	SF_RHO	SF_P	SF_T		
Type	F	F	F	F	F	F		
Default	1.0	1.0	1.0	1.0	1.0	1.0		
Remarks	2	2	2	2	2	2		

VARIABLE	DESCRIPTION
MSURFID	A mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID
N1, N2, …	Node ID's defining a segment
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID.
LC_U	Load curve ID to describe the x-component of the velocity versus time; see *DEFINE_CURVE.

VARIABLE	DESCRIPTION
LC_V	Load curve ID to describe the y-component of the velocity versus time.
LC_W	Load curve ID to describe the z-component of the velocity versus time.
LC_RHO	Load curve ID to describe the density versus time.
LC_P	Load curve ID to describe the pressure versus time.
LC_T	Load curve ID to describe the temperature versus time.
SF_U	Scale factor for LC_U (default = 1.0).
SF_V	Scale factor for LC_V (default = 1.0).
SF_W	Scale factor for LC_W (default = 1.0).
SF_RHO	Scale factor for LC_RHO (default = 1.0).
SF_P	Scale factor for LC_P (default = 1.0).
SF_T	Scale factor for LC_T (default = 1.0).

Remarks:

1. On each centroid or set of centroids, the variables (v_x, v_y, v_z, ρ, P, T) that are given values must be consistent and make the model well-posed (i.e., be such that the solution of the model exists, is unique and physical).

2. If any of the load curves are 0, the corresponding variable will take the constant value of the corresponding scale factor. For instance, if LC_RHO = 0, then the constant value of the density for this boundary condition will be SF_RHO.

3. If a load ID is -1 for a given variable, then the boundary value for that variable is computed by the solver, and not specified by the user.

CESE_BOUNDARY_PRESCRIBED_VN_*OPTION

Available options include:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: For the CESE compressible flow solver, set boundary values for velocity, density, pressure and temperature. Boundary values are applied at the centroid of elements connected with this boundary. OPTION = SET and OPTION = SEGMENT are for user defined meshes whereas OPTION = MSURF or MSURF_SET are associated with the automatic volume mesher (See *MESH keywords).

That is, the MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.

2. Card 2 reads in load curve IDs.

3. Card 3 reads in scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID	IDCOMP						
Type	I	I						
Default	none	none						

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S	IDCOMP						
Type	I	I						
Default	none	none						

Set Card. Card 1 format used when the SET keyword option is active.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID	IDCOMP						
Type	I	I						
Default	none	none						

Segment Card. Card 1 for SEGMENT keyword option is active.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N1	N3	N4	IDCOMP			
Type	I	I	I	I	I			
Default	none	none	none	none	none			

Load Curve Card.

Card 2	1	2	3	4	5	6	7	8
Variable	LC_VN			LC_RHO	LC_P	LC_T		
Type	I			I	I	I		
Remarks	1,2,3			1,2,3	1,2,3	1,2,3		

Scale Factor Card.

Card 3	1	2	3	4	5	6	7	8
Variable	SF_VN			SF_RHO	SF_P	SF_T		
Type	F			F	F	F		
Default	1.0			1.0	1.0	1.0		
Remarks	2			2	2	2		

VARIABLE	DESCRIPTION
MSURFID	A mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID
N1, N2, …	Node ID's defining a segment
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID.
LC_VN	Load curve ID to describe the normal velocity versus time; see *DEFINE_CURVE.

VARIABLE	DESCRIPTION
LC_RHO	Load curve ID to describe the density versus time.
LC_P	Load curve ID to describe the pressure versus time.
LC_T	Load curve ID to describe the temperature versus time.
SF_VN	Scale factor for LC_VN (default = 1.0).
SF_RHO	Scale factor for LC_RHO (default = 1.0).
SF_P	Scale factor for LC_P (default = 1.0).
SF_T	Scale factor for LC_T (default = 1.0).

Remarks:

1. On each centroid or set of centroids, the variables (V_N, ρ, P, T) that are given values must be consistent and make the model well-posed (i.e., be such that the solution of the model exists, is unique and physical).

2. If any of the load curves are 0, the corresponding variable will take the constant value of the corresponding scale factor. For instance, if LC_RHO = 0, then the constant value of the density for this boundary condition will be SF_RHO.

3. If a load ID is -1 for a given variable, then the boundary value for that variable is computed by the solver, and not specified by the user.

***CESE_BOUNDARY_REFLECTIVE**_*OPTION*

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Define a reflective boundary condition (RBC) for the CESE compressible flow solver. This boundary condition can be applied on a symmetrical surface or a solid wall of the computational domain.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Surface Part Card. Card 1 format used when the MSURF keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID							
Type	I							
Default	none							

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S							
Type	I							
Default	none							

Set Card. Card 1 format used when the SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

Segment Cards. Card 1 format used when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID.
N1, N2, …	Node IDs defining a segment

Remarks:

1. This boundary condition has the same effect as a solid-wall boundary condition for inviscid flows.

***CESE_BOUNDARY_SLIDING**_*OPTION*

Available options are:

MSURF

MSURF_SET

SET

SEGMENT

Purpose: Allows nodes of a fluid surface to translate in the main direction of mesh movement. This is useful in piston type applications.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Surface Part Card. Card 1 format used when the MSURF keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID							
Type	I							
Default	none							

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S							
Type	I							
Default	none							

Set Card. Card 1 format used when the SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

Segment Cards. Card 1 format used when SEGMENT keyword option is active. Include an additional card for each corresponding pair of segments. This input ends at the next keyword ("*") card.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID
N1, N2, …	Node IDs defining a segment

***CESE_BOUNDARY_SOLID_WALL**_OPTION1_OPTION2

For *OPTION1* the choices are:

MSURF

MSURF_SET

SET

SEGMENT

For *OPTION2* the choices are:

<BLANK>

ROTAT

Purpose: Define a solid wall boundary condition (SBC) for this CESE compressible flow solver. This boundary condition can be applied at a solid boundary that is the physical boundary for the flow field. For inviscid flow, this will be a slip boundary condition; while for viscous flows, it is a no-slip boundary condition.

The MSURF and MSURF_SET options are used when the CESE mesh has been created using *MESH cards. The SET and SEGMENT cards are used when *ELEMENT_SOLID cards are used to define the CESE mesh.

Card Sets. The following sequence of cards comprises a *single set*. LS-DYNA will continue reading *CESE_BOUNDARY_SOLID_WALL card sets until the next keyword ("*") card is encountered.

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSURFID	LCID	Vx	Vy	Vz	Nx	Ny	Nz
Type	I	I	F	F	F	F	F	F
Default	none	0	0.0	0.0	0.0	0.0	0.0	0.0
Remarks		2, 3	2	2	2	3	3	3

Surface Part Set Card. Card 1 format used when the MSURF_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSURF_S	LCID	Vx	Vy	Vz	Nx	Ny	Nz
Type	I	I	F	F	F	F	F	F
Default	none	0	0.0	0.0	0.0	0.0	0.0	0.0
Remarks		2, 3	2	2	2	3	3	3

Set Card. Card 1 format used when the SET keyword option is active.

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID	LCID	Vx	Vy	Vz	Nx	Ny	Nz
Type	I	I	F	F	F	F	F	F
Default	none	0	0.0	0.0	0.0	0.0	0.0	0.0
Remarks		2, 3	2	2	2	3	3	3

Segment Card. Card 1 format used when SEGMENT keyword option is active.

Card 1d	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4	LCID	Vx	Vy	Vz
Type	I	I	I	I	I	F	F	F
Default	none	none	none	none	0	0.0	0.0	0.0
Remarks					2, 3	2	2	2

Rotating Axis Card. Additional card for the "Segment Card" case that is read when the ROTAT keyword option is used.

Card 2	1	2	3	4	5	6	7	8
Variable	Nx	Ny	Nz					
Type	F	F	F					
Default	0.0	0.0	0.0					
Remarks	3	3	3					

VARIABLE	DESCRIPTION
MSURFID	Mesh surface part ID referenced in *MESH_SURFACE_ELEMENT cards.
MSURF_S	Identifier of a set of mesh surface part IDs created with a *LSO_-ID_SET card, where each mesh surface part ID in the set is referenced in *MESH_SURFACE_ELEMENT cards.
SSID	Segment set ID
N1, N2, …	Node ID's defining a segment
LCID	Load curve ID to define this solid wall boundary movement

If OPTION2 = <BLANK>:

Vx, Vy, Vz	velocity vector of the solid wall: LCID.EQ.0: it is defined by (Vx, Vy, Vz) itself; LCID.NE.0: it will be defined by both of the load curve and (Vx, Vy, Vz); Nx, Ny, Nz are not used in this case.

If OPTION2 = ROTAT:

Vx, Vy, Vz	x-,y- & z-coordinates of a point on the rotating axis
Nx, Ny, Nz	Unit vector of the rotating axis (for the 2D case, this is not used). The rotating frequency (Hz) is given by the load curve.

Remarks:

1. In this solid-wall condition (SBC), the boundary movement can only be in the tangential direction of the wall and should not affect the fluid domain size and mesh during the calculation, otherwise an FSI or moving mesh solver should be used. Also, this moving SBC only affects viscous flows (no-slip BC).

2. If LCID = 0 and Vx = Vy = Vz = 0.0 (default), this will be a regular solid wall BC.

3. For rotating SBC, LCID > 0 must be used to define the rotating speed frequency (Hz). Also, in the 2D case, (Nx, Ny, Nz) does not need to be defined because it is not needed.

*CESE_CHEMISTRY_D3PLOT

Purpose: Cause mass fractions of the listed chemical species to be added to the CESE d3plot output. This is only used when chemistry is being solved with the CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	MODELID							
Type	I							
Default	none							

Species Cards. Include one card for each species to be included in the d3plot database. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	SPECIES							
Type	A							

VARIABLE	DESCRIPTION
MODELID	Identifier of a Chemkin-compatible chemistry model.
SPECIES	Name of a chemical species that is defined in the chemistry model identified by MODELID (see *CHEMISTRY_MODEL).

*CESE_CONTROL_LIMITER

Purpose: Sets some stability parameters used in the CESE scheme for this CESE compressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	IDLMT	ALFA	BETA	EPSR				
Type	I	F	F	F				
Default	0	0.0	0.0	0.0				
Remarks		1	2	3				

VARIABLE	DESCRIPTION
IDLMT	Set the stability limiter option (See CESE theory manual): EQ.0: limiter format 1 (Re-weighting). EQ.1: limiter format 2 (Relaxing).
ALFA	Re-weighting coefficient (See CESE theory manual)
BETA	Numerical viscosity control coefficient (See CESE theory manual)
EPSR	Stability control coefficient (See CESE theory manual)

Remarks:

1. $\alpha \geq 0$; larger values give more stability, but less accuracy. Usually $\alpha = 2.0$ or 4.0 will be enough for normal shock problems.

2. $0 \leq \beta \leq 1$; larger values give more stability. For problems with shock waves, $\beta = 1.0$ is recommended.

3. $\varepsilon \geq 0$; larger values give more stability, but less accuracy.

*CESE_CONTROL_MESH_MOV

Purpose: For moving mesh CESE, this keyword is used to choose the type of algorithm to be used for calculating mesh movement.

Card 1	1	2	3	4	5	6	7	8
Variable	MMSH	LIM_ITER	RELTOL					
Type	I	I	F					
Default	1	100	1.0e-3					

VARIABLE	DESCRIPTION
MMSH	Mesh motion selector: EQ.1: mesh moves using an implicit ball-vertex spring method. EQ.9: the IDW scheme is used to move the mesh.
LIM_ITER	Maximum number of linear solver iterations for the ball-vertex linear system.
RELTOL	Relative tolerance to use as a stopping criterion for the iterative linear solver (conjugate gradient solver with diagonal scaling preconditioner).

.

*CESE_CONTROL_SOLVER

Purpose: Set general purpose control variables for the CESE compressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	ICESE	IFLOW	IGEOM	IFRAME	MIXID	IDC	ISNAN	
Type	I	I	I	I	I	F	I	
Default	0	0	none	0	none	0.25	0	
Remarks			1			2		

VARIABLE	DESCRIPTION
ICESE	Sets the framework of the CESE solver. EQ.0: Fixed Eulerian EQ.100: Moving Mesh FSI EQ.200: Immersed boundary FSI
IFLOW	Sets the compressible flow types: EQ.0: Viscous flows (laminar) EQ.1: Inviscid flows
IGEOM	Sets the geometric dimension: EQ.2: Two-dimensional (2D) problem EQ.3: Three-dimensional (3D) problem EQ.101: 2D axisymmetric
IFRAME	Choose the frame of reference: EQ.0: Usual non-moving reference frame (default) EQ.1000: Non-inertial rotating reference frame
MIXID	Chemistry model ID that defines the chemical species to include in the mixing model (see *CHEMISTRY_MODEL). The species information is given through the model's card specifying the Chemkin-compatible input.

VARIABLE	DESCRIPTION
IDC	Contact interaction detection coefficient (for FSI and conjugate heat transfer problems).
ISNAN	Flag to check for a NaN in the CESE solver solution arrays at the completion of each time step. This option can be useful for debugging purposes. There is a cost overhead when this option is active. EQ.0: No checking EQ.1: Checking is active.

Remarks:

1. **2D Solvers and Mesh Geometry.** For the 2D (IGEOM = 2) or 2D axisymmetric (IGEOM = 101) solver, the mesh should only be distributed in the *x*-*y* plane with the boundary conditions given only at the *x*-*y* domain boundaries. Otherwise, a warning message will be given, and the 3D solver will be triggered instead.

   The 2D axisymmetric case will work only if the 2D mesh and corresponding boundary conditions are properly defined, with the *x* and *y* coordinates corresponding to the radial and axial directions, respectively.

2. **Contact Interaction Detection Coefficient.** IDC is the same type of variable that is input on the *ICFD_CONTROL_FSI card. For an explanation, see Remark 1 for the *ICFD_CONTROL_FSI card.

3. **CESE Solver and Restarts.** Restarts are supported for the CESE solver. However, restarts do not currently work for simulations with conjugate heat transfer.

*CESE_CONTROL_TIMESTEP

Purpose: Sets the time-step control parameters for the CESE compressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	IDDT	CFL	DTINT					
Type	I	F	F					
Default	0	0.9	10^{-3}					

VARIABLE	DESCRIPTION
IDDT	Sets the time step option: EQ.0: fixed time step size (DTINT, meaning the given initial time step size) NE.0: the time step size will be calculated based on the given CFL-number and the flow solution at the previous time step.
CFL	CFL number (Courant–Friedrichs–Lewy condition) ($0.0 < CFL \leq 1.0$)
DTINT	Initial time step size

*CESE_DATABASE_ELOUT

Purpose: Enable the output of CESE data on elements. If more than one element set is defined, then several output files will be generated.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Element Sets Card. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	ELSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be created: EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the CESE timestep will be used.
ELSID	Solid elements set ID

Remarks:

1. **Database File Name.** The file name for this database is cese_elout.dat.

2. **Restrictions.** The *CESE_DATABASE… cards (pointout/elout) are restricted to CFD-only problems. They do not give correct results for chemically reacting flow where *CESE couples with *CHEMISTRY.

*CESE_DATABASE_FLUXAVG

Purpose: Enable the output of CESE data on segment sets. If more than one segment set is defined, then several output files will be generated.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Segment Sets Card. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be created: EQ.0: No output file is generated. EQ.1: The output file giving the average fluxes is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the CESE time step will be used.
SSID	Segment set ID

Remarks:

1. **Database File Name.** The file name for this database is cese_fluxavg.dat.

2. **Restrictions.** The *CESE_DATABASE… cards (pointout/elout/…) are restricted to CFD-only problems. They do not give correct results for chemically reacting flow where *CESE couples with *CHEMISTRY.

*CESE_DATABASE_FSIDRAG

Purpose: Enable the output of the total fluid pressure force applied to solid parts in FSI problems at every time step.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be created: EQ.0: No output file is generated. EQ.1: The output file giving the pressure forces is generated.

Remarks:

1. **Database File Names.** The file names for this database are cese_dragsol.dat, cese_dragshell.dat, cese_dragsol2D.dat and cese_dragbeam.dat, depending on what kind of solid is used.

*CESE_DATABASE_POINTOUT

Purpose: Enable the output of CESE data at points.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	PSID	DTOUT	PSTYPE	VX	VY	VZ		
Type	I	F	I	F	F	F		
Default	0	0.	0	0.	0.	0.		

Point Specification Card. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	PID	X	Y	Z				
Type	I	F	F	F				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
PSID	Point set ID
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the CESE timestep will be used.
PSTYPE	Point set type: EQ.0: Fixed points EQ.1: Tracer points using prescribed velocity EQ.2: Tracer points using fluid velocity
VX, VY, VZ	Constant velocities to be used when PSTYPE = 1
PID	Point ID

VARIABLE	DESCRIPTION
X, Y, Z	Initial coordinates of the point

Remarks:

1. **Database File Name.** The file name for this database is cese_pointout.dat.

2. **Restrictions.** The *CESE_DATABASE… cards (pointout/elout) are restricted to CFD-only problems. They do not give correct results for chemically reacting flow where *CESE couples with *CHEMISTRY.

*CESE_DATABASE_SSETDRAG

Purpose: Enable the output of CESE drag forces on segment sets. If more than one segment set is specified, then several output files will be generated.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Segment Sets Card. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be created: EQ.0: No output file is generated. EQ.1: The output file giving the average fluxes is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the CESE timestep will be used.
SSID	Segment set ID

Remarks:

1. **Database File Name.** The file name for this database is cese_ssetdrag.dat.

2. **Friction Drag Results and Mesh Size.** For the friction drag to give consistent results, special care must be given to the mesh close to the solid wall boundary (good capturing of the boundary layer behavior). We recommend a very fine structured mesh.

*CESE_DEFINE_NONINERTIAL

Purpose: Define the CESE problem domain as a non-inertial rotating frame that rotates at a constant rate. This is used in rotating problems such as spinning cylinders, wind turbines and turbo machinery.

Card 1	1	2	3	4	5	6	7	8
Variable	FREQ	LCID	PID	Nx	Ny	Nz		
Type	F	I	I	F	F	F		
Default	none	0	none	none	none	none		

Card 2	1	2	3	4	5	6	7	8
Variable	L	R	RELV					
Type	F	F	I					
Default	none	none	0					

VARIABLE	DESCRIPTION
FREQ	Frequency of rotation.
LCID	Load curve ID for scaling factor of FREQ.
PID	Starting point ID for the reference frame (See *CESE_DEFINE_POINT).
Nx, Ny, Nz	Rotating axis direction.
L	Length of rotating frame.
R	Radius of rotating frame.

VARIABLE	DESCRIPTION
RELV	Velocity display mode: EQ.0: Relative velocity, only the non-rotating components of the velocity are output. EQ.1: Absolute velocity is output.

*CESE_DEFINE_POINT

Purpose: Define points to be used by the CESE solver.

Point Cards. Include one card for each point. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	NID	X	Y	Z				
Type	I	F	F	F				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
NID	Identifier for this point.
X, Y, Z	Coordinates of the point.

*CESE_DRAG

Purpose: Provide the far-field (or free-stream) fluid pressure.

Card 1	1	2	3	4	5	6	7	8
Variable	PRESS							
Type	F							

VARIABLE	DESCRIPTION
PRESS	Value of the free-stream fluid pressure (in units used by the current problem).

*CESE_EOS_CAV_HOMOG_EQUILIB

Purpose: Define the coefficients in the equation of state (EOS) for the homogeneous equilibrium cavitation model.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	ρ_{vap}	ρ_{liq}	a_{vap}	a_{liq}	μ_{vap}	μ_{liq}	P_{SatVap}
Type	I	F	F	F	F	F	F	F
Default	none	0.8	880.0	334.0	1386.0	1.435e-5	1.586e-4	1.2e+4

VARIABLE	DESCRIPTION
EOSID	Equation of state identifier
ρ_{vap}	density of the saturated vapor
ρ_{liq}	density of the saturated liquid
a_{vap}	sound speed of the saturated vapor
a_{liq}	sound speed of the saturated liquid
μ_{vap}	dynamic viscosity of the vapor
μ_{liq}	dynamic viscosity of the liquid
P_{SatVap}	pressure of the saturated vapor

Remarks:

1. Once a cavitation EOS is used, the cavitation flow solver will be triggered.

2. In this homogeneous equilibrium cavitation model, a barotropic equation of state is used. This model can be used in small scale & high speed cavitation flows, and it is not good for large-scale, low-speed cavitation calculations.

*CESE_EOS_IDEAL_GAS

Purpose: Define the coefficients Cv and Cp in the equation of state for an ideal gas in the CESE fluid solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	Cv	Cp					
Type	I	F	F					
Default	none	717.5	1004.5					

VARIABLE	DESCRIPTION
EOSID	Equation of state identifier
Cv	Specific heat at constant volume
Cp	Specific heat at constant pressure

Remarks:

1. **Units.** As with other solvers in LS-DYNA, the user is responsible for unit consistency. For example, if a user wants to use dimensionless variables, Cv and Cp should also be replaced by the corresponding dimensionless ones.

*CESE_EOS_INFLATOR1

Purpose: To define an EOS using Cp and Cv thermodynamic expansions for an inflator gas mixture with a single temperature range.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID							
Type	I							
Default	none							

Card 2	1	2	3	4	5	6	7	8
Variable	Cp0	Cp1	Cp2	Cp3	Cp4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Card 3	1	2	3	4	5	6	7	8
Variable	Cv0	Cv1	Cv2	Cv3	Cv4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

VARIABLE	DESCRIPTION
EOSID	Equation of state identifier for the CESE solver.
Cp0, …, Cp4	Coefficients of temperature-dependent specific heat at constant pressure $C_p(T) = C_{p0} + C_{p1}\,T + C_{p2}\,T^2 + C_{p3}\,T^3 + C_{p4}\,T^4$

VARIABLE	DESCRIPTION
Cv0, …, Cv4	Coefficients of temperature-dependent specific heat at constant volume $C_v(T) = C_{v0} + C_{v1}\,T + C_{v2}\,T^2 + C_{v3}\,T^3 + C_{v4}\,T^4$

Remark:

1. These coefficient expansions for the specific heats over the entire temperature range are generated by the 0-D inflator model solver. See *CHEMISTRY_CONTROL_-INFLATOR and *CHEMISTRY_INFLATOR_PROPERTIES for details related to running that solver.

*CESE_EOS_INFLATOR2

Purpose: To define an EOS using Cp and Cv thermodynamic expansions for an inflator gas mixture with two temperature ranges, one below 1000 degrees Kelvin, and the other above 1000 degrees Kelvin.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID							
Type	I							
Default	none							

Card for the expansion of Specific Heat at Constant Pressure. Valid for T < 1000 ^{0}K

Card 2	1	2	3	4	5	6	7	8
Variable	Cp1_0	Cp1_1	Cp1_2	Cp1_3	Cp1_4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Card for the expansion of Specific Heat at Constant Pressure. Valid for T > 1000 ^{0}K.

Card 3	1	2	3	4	5	6	7	8
Variable	Cp2_0	Cp2_1	Cp2_2	Cp2_3	Cp2_4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Card for the expansion of Specific Heat at Constant Volume. Valid for T < 1000 ^{0}K

Card 4	1	2	3	4	5	6	7	8
Variable	Cv1_0	Cv1_1	Cv1_2	Cv1_3	Cv1_4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Card for the expansion of Specific Heat at Constant Volume. Valid for T > 1000 ^{0}K.

Card 5	1	2	3	4	5	6	7	8
Variable	Cv2_0	Cv2_1	Cv2_2	Cv2_3	Cv2_4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

VARIABLE	DESCRIPTION
EOSID	Equation of state identifier for the CESE solver.
Cp1_0, …, Cp1_4	Coefficients of temperature-dependent specific heat at constant pressure valid for T < 1000 ^{0}K. $C_{p1}(T) = C_{p1\_0} + C_{p1\_1}\,T + C_{p1\_2}\,T^2 + C_{p1\_3}\,T^3 + C_{p1\_4}\,T^4$
Cp2_0, …, Cp2_4	Coefficients of temperature-dependent specific heat at constant pressure valid for T > 1000 ^{0}K. $C_{p2}(T) = C_{p2\_0} + C_{p2\_1}\,T + C_{p2\_2}\,T^2 + C_{p2\_3}\,T^3 + C_{p2\_4}\,T^4$
Cv1_0, …, Cv1_4	Coefficients of temperature-dependent specific heat at constant volume valid for T < 1000 ^{0}K. $C_{v1}(T) = C_{v1\_0} + C_{v1\_1}\,T + C_{v1\_2}\,T^2 + C_{v1\_3}\,T^3 + C_{v1\_4}\,T^4$
Cv2_0, …, Cv2_4	Coefficients of temperature-dependent specific heat at constant volume valid for T > 1000 ^{0}K. $C_{v2}(T) = C_{v2\_0} + C_{v2\_1}\,T + C_{v2\_2}\,T^2 + C_{v2\_3}\,T^3 + C_{v2\_4}\,T^4$

Remark:

2. These coefficient expansions for the specific heats over two temperature ranges are generated by the 0-D inflator model solver. See *CHEMISTRY_CONTROL_INFLATOR and *CHEMISTRY_INFLATOR_PROPERTIES for details related to running that solver.

*CESE_FSI_EXCLUDE

Purpose: Provide a list of mechanics solver parts that are not involve in the CESE FSI calculation. This is intended to be used as an efficiency measure for parts that will not involve significant FSI interactions with the CESE compressible fluid solver..

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
PIDn	IDs of mechanics parts that will be excluded from the FSI interaction calculation with the CESE solver.

*CESE_INITIAL

Purpose: Specify constant initial conditions (ICs) for flow variables at the centroid of each fluid element.

Card 1	1	2	3	4	5	6	7	8
Variable	U	V	W	RH	P	T		
Type	F	F	F	F	F	F		
Default	0	0.0	0.0	1.225	0.0	0.0		

VARIABLE	DESCRIPTION
U, V, W	x-, y-, z-velocity components, respectively
RHO	Density, ρ
P	Pressure, P
T	Temperature, T

Remarks:

1. **Required Input.** Usually, only two of ρ, P, and T need to be specified (besides the velocity). If all three are given, only ρ and P will be used.

2. **Applicable Elements.** These initial conditions will be applied only in those elements that have not been assigned a value by *CESE_INITIAL_*OPTION* cards for individual elements or sets of elements.

***CESE_INITIAL**_OPTION

Available options include:

SET

ELEMENT

Purpose: Specify initial conditions for the flow variables at the centroid of each element in a set of elements or at the centroid of a single element.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	EID/ESID	U	V	W	RHO	P	T	
Type	I	F	F	F	F	F	F	
Default	none	0.0	0.0	0.0	1.225	0.0	0.0	
Remarks					1	1	1	

VARIABLE	DESCRIPTION
EID/ESID	Solid element ID (EID) or solid element set ID (ESID)
U, V, W	x-, y-, z-velocity components, respectively
RHO	Density, ρ
P	Pressure, P
T	Temperature, T

Remarks:

1. **Required Input.** Usually, only two of ρ, P, and T need to be specified (along with the velocity). If all three are given, only ρ and P will be used.

2. **Initial Condition Specification Priority.** The priority of this card is higher than *CESE_INITIAL, meaning that if an element is assigned an initial value by this card, *CESE_INITIAL will no longer apply to that element.

*CESE_INITIAL_CHEMISTRY

Purpose: Initializes the chemistry and fluid state in every element of the CESE mesh that has not already been initialized by one of the other *CESE_INITIAL_CHEMISTRY cards. This is only used when chemistry is being solved with the CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	CHEMID	COMPID						
Type	I	I						
Default	none	none						

Card 2	1	2	3	4	5	6	7	8
Variable	UIC	VIC	WIC	RHOIC	PIC	TIC	HIC	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
CHEMID	Identifier of chemistry control card to use
COMPID	Identifier of chemical composition to use
UIC	*X*-component of the fluid velocity
VIC	*Y*-component of the fluid velocity
WIC	*Z*-component of the fluid velocity
RHOIC	Initial fluid density
PIC	Initial fluid pressure
TIC	Initial fluid temperature

VARIABLE	DESCRIPTION
HIC	Initial fluid enthalpy. However, when CHEMID refers to a ZND 1-step reaction card, this is the progressive variable (degree of combustion).

*CESE_INITIAL_CHEMISTRY_ELEMENT

Purpose: Initializes the chemistry and fluid state in every element of the list of CESE elements. This is only used when chemistry is being solved with the CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	CHEMID	COMPID						
Type	I	I						
Default	none	none						

Card 2	1	2	3	4	5	6	7	8
Variable	UIC	VIC	WIC	RHOIC	PIC	TIC	HIC	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

Element List Card. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 3	1	2	3	4	5	6	7	8
Variable	ELE1	ELE2	ELE3	ELE4	ELE5	ELE6	ELE7	ELE8
Type	I	I	I	I	I	I	I	I

VARIABLE	DESCRIPTION
CHEMID	Identifier of chemistry control card to use
COMPID	Identifier of chemical composition to use
UIC	*X*-component of the fluid velocity
VIC	*Y*-component of the fluid velocity

VARIABLE	DESCRIPTION
WIC	Z-component of the fluid velocity
RHOIC	Initial fluid density
PIC	Initial fluid pressure
TIC	Initial fluid temperature
HIC	Initial fluid enthalpy. However, when CHEMID refers to a ZND 1-step reaction card, this is the progressive variable (degree of combustion).
ELE*i*	User element numbers to initialize

*CESE_INITIAL_CHEMISTRY_PART

Purpose: Initializes the chemistry and fluid state in every element of the specified CESE part that has not already been initialized by *CESE_INITIAL_CHEMISTRY_ELEMENT or *CESE_INITIAL_CHEMISTRY_SET cards. This is only used when chemistry is being solved with the CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	PARTID	CHEMID	COMPID					
Type	I	I	I					
Default	none	none	none					

Card 2	1	2	3	4	5	6	7	8
Variable	UIC	VIC	WIC	RHOIC	PIC	TIC	HIC	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
PARTID	Identifier of the CESE part on which to initialize.
CHEMID	Identifier of chemistry control card to use.
COMPID	Identifier of chemical composition to use.
UIC	X-component of the fluid velocity.
VIC	Y-component of the fluid velocity.
WIC	Z-component of the fluid velocity.
RHOIC	Initial fluid density.
PIC	Initial fluid pressure.
TIC	Initial fluid temperature.

VARIABLE	DESCRIPTION
HIC	Initial fluid enthalpy. However, when CHEMID refers to a ZND 1-step reaction card, this is the progressive variable (degree of combustion).

*CESE_INITIAL_CHEMISTRY_SET

Purpose: Initializes the chemistry and fluid state in every element of the specified element set in the CESE mesh that has not already been initialized by *CESE_INITIAL_CHEMISTRY_ELEMENT cards. This is only used when chemistry is being solved with the CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	SETID	CHEMID	COMPID					
Type	I	I	I					
Default	none	none	none					

Card 2	1	2	3	4	5	6	7	8
Variable	UIC	VIC	WIC	RHOIC	PIC	TIC	HIC	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
SETID	Identifier of the CESE element set to initialize.
CHEMID	Identifier of chemistry control card to use.
COMPID	Identifier of chemical composition to use.
UIC	X-component of the fluid velocity.
VIC	Y-component of the fluid velocity.
WIC	Z-component of the fluid velocity.
RHOIC	Initial fluid density.
PIC	Initial fluid pressure.
TIC	Initial fluid temperature.

VARIABLE	DESCRIPTION
HIC	Initial fluid enthalpy. However, when CHEMID refers to a ZND 1-step reaction card, this is the progressive variable (degree of combustion).

*CESE_MAT_000

Purpose: Define the fluid (gas) properties in a viscous flow for the CESE solver.

Material Definition Cards. Include one card for each instance of this material type. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MU	K					
Type	I	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
MID	Material identifier
MU	Fluid dynamic viscosity. For Air at 15 °C, MU = 1.81×10^{-5} $^{kg}/_{ms}$
K	Thermal conductivity of the fluid

Remarks:

1. The viscosity is only used viscous flows, so for inviscid flows, it is not necessary to define it. The thermal conductivity is only used to calculate the heat transfer between the structure and the thermal solver when coupling is activated.

2. As with other solvers in LS-DYNA, the user is responsible for unit consistency. For example, if dimensionless variables are used, MU should be replaced by the corresponding dimensionless one.

*CESE_MAT_001(_GAS)

Purpose: Define the fluid (gas) properties in a viscous flow for the CESE solver.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	C1	C2	PRND				
Type	I	F	F	F				
Default	none	1.458E-6	110.4	0.72				

VARIABLE	DESCRIPTION
MID	Material identifier
C1, C2	Two coefficients in the Sutherland's formula for viscosity, i.e., $$\mu = \frac{C_1 T^{\frac{3}{2}}}{T + C_2}$$ where C_1 and C_2 are constants for a given gas. For example, for air at moderate temperatures, $$C_1 = 1.458 \times 10^{-6}\,\mathrm{kg/msK^{1/2}}, \quad C_2 = 110.4\,\mathrm{K}$$
PRND	The Prandtl Number (used to determine the coefficient of thermal conductivity). It is approximately constant for most gases. For air at standard conditions PRND = 0.72.

Remarks:

1. C1 and C2 are only used to calculate the viscosity in viscous flows, so for inviscid flows, this material card is not needed. The Prandtl number is used to extract the thermal conductivity, which is used when thermal coupling with the structure is activated.

2. As with other solvers in LS-DYNA, the user is responsible for unit consistency. For example, if dimensionless variables are used, C_1 and C_2 should be replaced by the corresponding dimensionless ones.

*CESE_MAT_002

Purpose: Define the fluid (gas) properties in a viscous flow for the CESE solver.

Material Definition Cards. Include one card for each instance of this material type. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MU0	SMU	K0	SK	T0		
Type	I	F	F	F	F	F		
Default	none	1.716E-5	111.	0.0241	194.0	273.0		

VARIABLE	DESCRIPTION
MID	Material identifier
MU0 / SMU	Two coefficients appearing in the equation derived by combining Sutherland's formula with the power law for dilute gases: $$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{3/2} \frac{T_0 + S_\mu}{T + S_\mu} \ .$$ μ_0 is a reference value, and S_μ is an effective temperature called the Sutherland constant, which is characteristic of the gas. For air at moderate temperatures, $$\mu_0 = 1.716 \times 10^{-5}\,\mathrm{Ns/m^2}\,, \qquad S_\mu = 111\ \mathrm{K}$$
K0/SK	Two coefficients appearing in the equation derived by combining Sutherland's formula with the power law for dilute gases: $$\frac{k}{k_0} = \left(\frac{T}{T_0}\right)^{3/2} \frac{T_0 + S_k}{T + S_k} \ .$$ Here k is the thermal conductivity, k_0 is a reference value, and S_k is the Sutherland constant, which is characteristic of the gas. For air at moderate temperatures, $$k_0 = 0.0241\ \mathrm{W/m}\,, \qquad S_k = 194\ \mathrm{K}$$
T0	Reference temperature, T_0. The default value (273.0) is for air, in degrees K.

Remarks:

1. **Fields that Depend on Problem Physics.** The viscosity is only used for viscous flow. Therefore, for inviscid flows, it is not necessary to define it. The thermal conductivity is only used to calculate the heat transfer between the structure and the thermal solver when coupling is activated.

2. **Unit Consistency.** As with other solvers in LS-DYNA, the user is responsible for unit consistency. For example, if dimensionless variables are used, MU should be replaced by the corresponding dimensionless one.

*CESE_PART

Purpose: Define CESE solver parts, i.e., connect CESE material and EOS information.

Part Cards. Include one card for each CESE part. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	MID	EOSID					
Type	I	I	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
PID	Part identifier (must be different from any PID on a *PART card)
MID	Material identifier defined by a *CESE_MAT_… card
EOSID	Equation of state identifier defined by a *CESE_EOS_… card

Remarks:

1. Since material coefficients are only used in viscous flows, the MID can be left blank for inviscid flows.

*CESE_SURFACE_MECHSSID_D3PLOT

Purpose: Identify the surfaces to be used in generating surface D3PLOT output for the CESE solver. These surfaces must be on the outside of volume element parts that are in contact with the CESE fluid mesh. The variables in question are part of the CESE FSI solution process or of the CESE conjugate heat transfer solver.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	SurfaceLabel						
Type	I	A						
Default	none	none						

VARIABLE	DESCRIPTION
SSID	Mechanics solver segment set ID that is in contact with the fluid CESE mesh.
SurfaceLabel	Name to use in d3plot output to identify the SSID for the LSPP user.

*CESE_SURFACE_MECHVARS_D3PLOT

Purpose: List of variables to output on the surfaces designated by the segment set IDs given in the *CESE_SURFACE_MECHSSID_D3PLOT cards. Most of the allowed variables are defined only on the fluid-structure interface, and so the segment set IDs defining a portion of the fluid-structure interface must involve only segments (element faces) that are on the outside of volume element parts that are in contact with the CESE fluid mesh.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	Output Quantity							
Type	A							
Default	none							

VARIABLE	DESCRIPTION
Output Quantity	Descriptive phrase for the mechanics surface variable to output for the LSPP user. Output will be done on all SSIDs selected by the *CESE_SURFACE_MECHSSID_D3PLOT cards in the problem. Supported variables include: FLUID FSI FORCE FLUID FSI PRESSURE INTERFACE TEMPERATURE SOLID INTERFACE HEAT FLUX FLUID INTERFACE HEAT FLUX INTERFACE HEAT FLUX RATE SOLID INTERFACE DISPLACEMENT SOLID INTERFACE VELOCITY SOLID INTERFACE ACCELERATION Force, displacement, velocity, and acceleration are output as vector quantities. The rest of the variables are scalar quantities. The fluxes are in the normal direction to the fluid/structure interface, with the heat fluxes relative to the normal pointing into the structure.

*CHEMISTRY

The keyword *CHEMISTRY is used to access chemistry databases that include Chemkin-based descriptions of a chemical model, as well as to select a method of solving the model. The keyword cards in this section are defined in alphabetical order:

*CHEMISTRY_BATTERY†

*CHEMISTRY_COMPOSITION

*CHEMISTRY_CONTROL_0D

*CHEMISTRY_CONTROL_1D†

*CHEMISTRY_CONTROL_CSP

*CHEMISTRY_CONTROL_FULL

*CHEMISTRY_CONTROL_INFLATOR†

*CHEMISTRY_CONTROL_TBX

*CHEMISTRY_CONTROL_ZND†

*CHEMISTRY_DET_INITIATION†

*CHEMISTRY_INFLATOR_PROPERTIES†

*CHEMISTRY_MODEL

*CHEMISTRY_PATH

†: Card may be used only once in a given model

An additional option "_TITLE" may be appended to all *CHEMISTRY keywords. If this option is used, then an 80 character string is read as a title from the first card of that keyword's input. At present, LS-DYNA does not make use of the title. Inclusion of titles gives greater clarity to input decks.

The *CHEMISTRY_BATTERY card is only intended to be used in battery electrochemistry models that involve *BATTERY_... cards. Note that none of the comments below apply for this particular case.

*CHEMISTRY

In order to use one of the chemistry solvers, the input must include at least one *CHEMISTRY_MODEL card. For each spatial region containing a different chemical composition, at least one *CHEMISTRY_COMPOSITION card is required.

The *CHEMISTRY_CONTROL_0D card is intended to be used in a standalone fashion to verify the validity of a given chemistry model. This model includes the total number of species and all elementary reactions with their Arrhenius rate parameters. For instance, this solver could be used to check the induction time of the model.

The *CHEMISTRY_BLAST_INITIATION, *CHEMISTRY_CONTROL_1D, *CHEMISTRY_DET_INITIATION, and *CHEMISTRY_CONTROL_ZND cards are intended to provide a one-dimensional initialization to a 2D or 3D chemically-reacting flow.

In order to perform a full, general purpose chemistry calculation in 2D or 3D, the *CHEMISTRY_CONTROL_FULL card should be used.

The *CHEMISTRY_CONTROL_CSP card is an option for reducing the number of species and reactions that are used in a general purpose chemistry calculation. Other reduction mechanisms are planned for the future.

An airbag inflator model is available with *CHEMISTRY_CONTROL_INFLATOR along with *CHEMISTRY_INFLATOR_PROPERTIES and a chemistry model that is referenced via three chemical compositions. This involves zero-dimensional modeling, with pyrotechnic inflator, and cold and hot flow hybrid inflator options.

The *CHEMISTRY_CONTROL_TBX card is intended for use only in a stochastic particle model, where the *STOCHASTIC_TBX_PARTICLES card is used.

*CHEMISTRY_BATTERY

Purpose: For the battery electrochemistry solver (*BATTERY), this card identifies the files that define a Chemkin chemistry model for use with that solver.

Battery Reaction and Species Input File Card (Chemkin-compatible).

Card 1	1	2	3	4	5	6	7	8
Variable	FILE1							
Type	A							

Thermodynamics Database File Card.

Card 2	1	2	3	4	5	6	7	8
Variable	FILE2							
Type	A							

Transport Properties Database File Card.

Card 3	1	2	3	4	5	6	7	8
Variable	FILE3							
Type	A							

VARIABLE	DESCRIPTION
FILE1	Name of the file containing the Chemkin-compatible input
FILE2	Name of the file containing the chemistry thermodynamics database
FILE3	Name of the file containing the chemistry transport properties database

*CHEMISTRY_COMPOSITION

Purpose: Provides a general way to specify a chemical composition via a list of species mole numbers in the context of a Chemkin database model.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	MODELID						
Type	I	I						
Default	none	none						

Species List Card. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	MOLFR	SPECIES						
Type	F	A						
Default	none	none						

VARIABLE	DESCRIPTION
ID	A unique identifier among all chemistry compositions.
MODELID	Identifier of a Chemkin-compatible chemistry model.
MOLFR	The number of moles corresponding to the species named in the SPECIES field. But if used with a *STOCHASTIC_TBX_PARTICLES card, it is the molar concentration of the species (in units of moles/[length]^3, where "[length]" is the user's length unit).
SPECIES	The Chemkin-compatible name of a chemical species that is defined in the chemistry model identified by MODELID (see *CHEMISTRY_MODEL).

*CHEMISTRY_CONTROL_0D

Purpose: Performs a zero-dimensional isotropic chemistry calculation that operates standalone (does not call the CESE solver). This is for ISOBARIC or ISOCHORIC cases.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	COMPID	SOLTYP	PLOTDT	CSP_SEL			
Type	I	I	I	F	I			
Default	none	none	none	1.0e-6	0			
Remarks					1			

Card 2	1	2	3	4	5	6	7	8
Variable	DT	TLIMIT	TIC	PIC	RIC	EIC		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

CSP Parameters Card. Include cards for each chemical species in the following format when CSP_SEL.GT.0. This input ends at the next keyword ("*") card.

Card 3	1	2	3	4	5	6	7	8
Variable	AMPL	YCUT						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
ID	Identifier for this 0D computation.

VARIABLE	DESCRIPTION
COMPID	Chemical composition identifier of composition to use.
SOLTYP	Type of 0D calculation: EQ.1: Isochoric EQ.2: Isobaric
PLOTDT	Simulation time interval for output both to the screen and to the isocom.csv file. This file can be loaded into LS-PREPOST for curve plotting using the x-y plot facility.
CSP_SEL	CSP solver option: EQ.0: Do not use the CSP solver, and ignore the AMPL and YCUT parameters (default). GT.0: Use the CSP solver, with the AMPL and YCUT parameters.
DT	Initial time step
TLIMIT	Time limit for the simulation
TIC	Initial temperature
PIC	Initial pressure
RIC	Initial density
EIC	Initial internal energy
AMPL	Relative accuracy for the mass fraction of a chemical species in the Chemkin input file.
YCUT	Absolute accuracy for the mass fraction of a chemical species in the Chemkin input file.

Remarks:

1. If CSP_SEL.GT.0, then instead of using the full chemistry solver, the computational singular perturbation (CSP) method solver is used.

*CHEMISTRY_CONTROL_1D

Purpose: Loads a previously-computed one-dimensional detonation. It is then available for use in the CESE solver for initializing a computation. In the product regions, this card overrides the initialization of the *CESE_INITIAL_CHEMISTRY_… cards.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	XYZD	DETDIR	CSP_SEL				
Type	I	F	I	I				
Default	none	none	none	0				
Remarks				1				

One-Dimensional Solution LSDA Input File Card.

Card 2	1	2	3	4	5	6	7	8
Variable	FILE							
Type	A							

CSP Parameters Card Include cards for each chemical species in the following format when CSP_SEL > 0. This input ends at the next keyword ("*") card.

Card 3	1	2	3	4	5	6	7	8
Variable	AMPL	YCUT						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
ID	Identifier for this one-dimensional detonation solution.
XYZD	Position of the detonation front in the DETDIR direction.

VARIABLE	DESCRIPTION
DETDIR	Detonation propagation direction EQ.1: x EQ.2: y EQ.3: z
CSP_SEL	CSP solver option: EQ.0: Do not use the CSP solver, and ignore the AMPL and YCUT parameters (default). GT.0: Use the CSP solver, with the AMPL and YCUT parameters.
FILE	Name of the LSDA file containing the one-dimensional solution.
AMPL	Relative accuracy for the mass fraction of a chemical species in the chemkin input file.
YCUT	Absolute accuracy for the mass fraction of a chemical species in the chemkin input file.

Remarks:

1. If CSP_SEL > 0, then instead of using the full chemistry solver, the computational singular perturbation (CSP) method solver is used.

*CHEMISTRY_CONTROL_CSP

Purpose: Computes reduced chemistry for a specified Chemkin chemistry model using the Computational Singular Perturbation (CSP) method. This card can be used for general-purpose chemical reaction calculations.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	IERROPT						
Type	I	I						
Default	none	none						

CSP Parameters Card. Include cards for each chemical species in the following format as indicated by the value of IERROPT. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	AMPL	YCUT						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
ID	Identifier for this computational singular perturbation solver.
IERROPT	Selector: EQ.0: AMPL and YCUT values for all chemical species are required. EQ.1: One CSP Parameter Card should be provided, and it will be used for all species.
AMPL	Relative accuracy for the mass fraction of a chemical species in the Chemkin input file.
YCUT	Absolute accuracy for the mass fraction of a chemical species in the Chemkin input file.

*CHEMISTRY_CONTROL_FULL

Purpose: Computes the full chemistry specified by a Chemkin chemistry model. This card can be used for general-purpose chemical reaction calculations.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	ERRLIM	RHOMIN	TMIN				
Type	I	F	F	F				
Default	none	none	0.0	0.0				

VARIABLE	DESCRIPTION
ID	Identifier for this full chemistry calculation.
ERRLIM	Error tolerance for the full chemistry calculation.
RHOMIN	Minimum fluid density above which chemical reactions are computed.
TMIN	Minimum temperature above which chemical reactions are computed.

*CHEMISTRY_CONTROL_INFLATOR

Purpose: Provide the required properties of an inflator model for airbag inflation.

Card 1	1	2	3	4	5	6	7	8
Variable	MODEL	OUT_TYPE	TRUNTIM	DELT	PTIME			
Type	I	I	F	F	F			
Remarks	1	2,4						

Inflator Output Database File (an ASCII file) Card.

Card 2	1	2	3	4	5	6	7	8
Variable	FILE							
Type	A							

Densities for Condensed Species. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 3	1	2	3	4	5	6	7	8
Variable	DENSITY	Species Name						
Type	F	A						
Default	none	none						
Remark		3						

VARIABLE	DESCRIPTION
MODEL	Type of inflator model to compute. EQ.1: Pyrotechnic model EQ.2: Hybrid model with cold flow option in the gas chamber EQ.3: Hybrid model with heat flow in the gas chamber EQ.4: Hybrid model with heat flow in one additional gas chamber EQ.5: Hybrid model with heat flow in two additional gas chambers
OUT_TYPE	Selects the output file format that will be used in an airbag simulation. EQ.0: Screen output calibration output (see Remark 4) EQ.1: CESE compressible flow solver (default) EQ.2: ALE solver EQ.3: CPM solver (with 2nd-order expansion of C_p) EQ.4: CPM solver (with 4th-order expansion of C_p)
TRUNTIM	Total run time.
DELT	Delta(t) to use in the model calculation.
PTIME	Time interval for output of time history data to FILE.
FILE	Name of the ASCII file in which to write the time history data and other data output by the inflator simulation.
DENSITY	Density of a condensed-phase species present in the inflator.
Species Name	Chemkin-compatible name of a condensed-phase species.

Remarks:

1. If MODEL = 3, the solution of an elementary reaction system is required for the finite-rate chemistry in the gas chamber. For MODEL = 4 and 5, the condensed phase is computed only in the combustion chamber.

2. Output file includes all of the necessary thermodynamics variables and load curves for the species mass flow rate, temperature, and density curve. This will make it possible to generate the velocity curve which is required by each solver that carries out an airbag simulation.

3. At least one of these cards will be input if condensed-phase species are present during the propellant combustion. In this case, the user must specify each condensed-phase density. This density is then used to compute the volume fractions in both the combustion and gas chamber, where the energy equations are needed.

4. If OUT_TYPE = 0, the propellant information will be displayed on the screen, including total mass, remaining mass percentage, and mass burning rate versus time, and the calibration data will be saved in the output file, including the time versus pressure, temperature, total mass flow rate, and individual species mass fractions for all chambers. With this option, the user can quickly see the effect of changing the parameters on the first three *CHEMISTRY_INFLATOR_PROPERTIES cards.

*CHEMISTRY_CONTROL_TBX

Purpose: Specify a chemistry solver for use in conjunction with stochastic TBX particles. This is intended only for modeling the second phase of an explosion where the explosive has embedded metal (aluminum) particles that are too large to have burned in the first phase of the explosion.

This chemistry card points to a *CHEMISTRY_MODEL card (via IDCHEM) with its associated *CHEMISTRY_COMPOSITION cards to set up the initial conditions. That is, it establishes the spatial distribution of the species in the model.

It is assumed that there is no chemical reaction rate information in the chemistry model files. This is done since a special chemical reaction mechanism is implemented for TBX modeling. If particles other than solid aluminum particles are embedded in the explosive, then another burn model has to be implemented.

Surface Part Card. Card 1 format used when the PART keyword option is active**.**

Card 1	1	2	3	4	5	6	7	8
Variable	IDCHEM	USEPAR						
Type	I	I						
Default	none	1						

VARIABLE	DESCRIPTION
IDCHEM	Identifier for this chemistry solver.
USEPAR	Coupling flag indicating if a *STOCHASTIC_TBX_PARTICLES card is provided for this model: EQ.1: uses a *STOCHASTIC_TBX_PARTICLES card (default). EQ.0: does not use such a card.

*CHEMISTRY_CONTROL_ZND

Purpose: Computes the one-dimensional reduced chemistry of a ZND model. It is then used in the initialization of the chemistry part of the CESE solver. When this card is used, the *CESE_INITIAL_CHEMISTRY… cards must specify the progressive variable (degree of combustion) in the HIC field.

Card 1	1	2	3	4	5	6	7	8
Variable	ID							
Type	I							
Default	none							

Card 2	1	2	3	4	5	6	7	8
Variable	F	EPLUS	Q0	GAM	XYZD	DETDIR		
Type	F	F	F	F	F	I		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
ID	Identifier for this full chemistry calculation.
F	Overdriven factor
EPLUS	EPLUS parameter of the ZND model.
Q0	Q0 parameter of the ZND model.
GAM	GAM parameter of the ZND model.
XYZD	Position of the detonation front in the DETDIR direction.
DETDIR	Detonation propagation direction (1 => X; 2 => Y; 3 => Z)

*CHEMISTRY_DET_INITIATION

Purpose: Performs a one-dimensional detonation calculation based upon a chemical composition and initial conditions. It is then available for use immediately in the CESE solver for initializing a computation, or it can be subsequently used by the *CHEMISTRY_CONTROL_1D card in a later run. In the product regions, this card overrides the initialization of the *CESE_INITIAL_CHEMISTRY… cards.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	COMPID	NMESH	DLEN	CFL	TLIMIT	XYZD	DETDIR
Type	I	I	I	F	F	F	F	I
Default	none	none	none	none	none	none	none	none

LSDA Output File Card.

Card 2	1	2	3	4	5	6	7	8
Variable	FILE							
Type	A							

VARIABLE	DESCRIPTION
ID	Identifier for this one-dimensional detonation computation.
COMPID	Chemical composition identifier of composition to use.
NMESH	Number of equal-width elements in the one-dimensional domain.
DLEN	Length of the one-dimensional domain.
CFL	Time-step limiting factor.
TLIMIT	Time limit for the simulation
XYZD	Position of the detonation front in the DETDIR direction.
DETDIR	Detonation propagation direction (1 => X; 2 => Y; 3 => Z)

VARIABLE	DESCRIPTION
FILE	Name of the LSDA file in which to write the one-dimensional solution.

*CHEMISTRY_INFLATOR_PROPERTIES

Purpose: Provide the required properties of an inflator model.

Card 1	1	2	3	4	5	6	7	8
Variable	COMP_ID	PDIA	PHEIGHT	PMASS	TOTMASS			
Type	I	F	F	F	F			
Remarks	1	2	2					

Card 2	1	2	3	4	5	6	7	8
Variable	TFLAME	PINDEX	A0	TDELAY	RISETIME			
Type	F	F	F	F	F			
Default	none	none	none	none	None			

Combustion Chamber Parameter Card.

Card 3	1	2	3	4	5	6	7	8
Variable	COMP1ID	VOL1	AREA1	CD1	P1	T1	DELP1	DELTI
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Gas Plenum Parameter Card.

Card 4	1	2	3	4	5	6	7	8
Variable	COMP2ID	VOL2	AREA2	CD2	P2	T2	DELP2	DELT2
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Tank Parameter Card.

Card 5	1	2	3	4	5	6	7	8
Variable	COMP3ID	VOL3	P3	T3				
Type	I	F	F	F				
Default	none	none	none	none				

Gas Chamber 1 (Optional, see Remark 3) Card.

Card 6	1	2	3	4	5	6	7	8
Variable	COMP4ID	VOL4	AREA4	CD4	P4	T4	DELP4	DELT4
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Gas Chamber 2 (Optional, see Remark 3) Card.

Card 7	1	2	3	4	5	6	7	8
Variable	COMP5ID	VOL5	AREA5	CD5	P5	T5	DELP5	DELT5
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
COMP_ID	Chemical composition identifier of the composition for the steady-state propellant combustion (see Remark 1).
PDIA	Propellant diameter (see Remark 2).
PHEIGHT	Propellant height (see Remark 2).
PMASS	Individual cylinder (or sphere) propellant mass.
TOTMASS	Total propellant mass.
TFLAME	Adiabatic flame (combustion) temperature.
PINDEX	Power of the pressure in rate of burn model.
A0	Steady-state constant.
TDELAY	Ignition time delay.
RISETIME	Rise time.
COMP1ID	Chemical composition identifier of composition to use in the combustion chamber.
VOL1	Volume of the combustion chamber.
AREA1	Area of the combustion chamber.
CD1	Discharge coefficient of the combustion chamber.
P1	Pressure in the combustion chamber.

VARIABLE	DESCRIPTION
T1	Temperature in the combustion chamber.
DELP1	Rupture pressure in the combustion chamber.
DELT1	Elapsed time for breaking the burst disk between the chambers
COMP2ID	Chemical composition identifier of composition to use in the gas plenum.
VOL2	Volume of the gas plenum.
AREA2	Area of the gas plenum.
CD2	Discharge coefficient of the gas plenum.
P2	Pressure in the gas plenum.
T2	Temperature in the gas plenum.
DELP2	Rupture pressure in the gas plenum.
DELT2	Elapsed time for breaking the burst disk between the chambers
COMP3ID	Chemical composition identifier of composition to use in the tank.
VOL3	Volume of the tank.
P3	Pressure in the tank.
T3	Temperature in the tank.
COMP4ID	Chemical composition identifier of composition to use in the additional (second) gas chamber.
VOL4	Volume of the second gas chamber.
P4	Pressure in the second gas chamber.
T4	Temperature in the second gas chamber.
DELP4	Rupture pressure in the second gas chamber.
DELT4	Elapsed time for breaking the burst disk between the first and second gas chambers
COMP5ID	Chemical composition identifier of composition to use in the additional (third) gas chamber.

VOL5	Volume of the third gas chamber.
P5	Pressure in the third gas chamber.
T5	Temperature in the third gas chamber.
DELP5	Rupture pressure in the third gas chamber.
DELT5	Elapsed time for breaking the burst disk between the second and third gas chambers

Remarks:

1. The propellant composition can be obtained by running a chemical equilibrium program such as NASA CEA, the CHEETAH code, or the PEP code. LSTC provides a modified version of the PEP code along with documentation for users; it is available upon request.

2. A spherical shape for the propellant particles can be chosen if an identical value for the diameter and height is given.

3. To simulate a 4 or 5 chamber inflator, an additional chamber card can be used. In these cases of the inflator models, the condensed phase species are limited to the combustion chamber only if involved in the propellant combustion.

*CHEMISTRY_MODEL

Purpose: Identifies the files that define a Chemkin chemistry model.

Card 1	1	2	3	4	5	6	7	8
Variable	MODELID	JACSEL	ERRLIM					
Type	I	I	F					
Default	none	1	1.0e-3					

Chemkin Input File Card.

Card 2	1	2	3	4	5	6	7	8
Variable	FILE1							
Type	A							

Thermodynamics Database File Card.

Card 3	1	2	3	4	5	6	7	8
Variable	FILE2							
Type	A							

Transport Properties Database File Card.

Card 4	1	2	3	4	5	6	7	8
Variable	FILE3							
Type	A							

VARIABLE	DESCRIPTION
MODELID	Identifier for this Chemkin-based chemistry model..

VARIABLE	DESCRIPTION
JACSEL	Selects the form of the Jacobian matrix for use in the source term. EQ.1: Fully implicit (default) EQ.2: Simplified implicit
ERRLIM	Allowed error in element balance in a chemical reaction.
FILE1	Name of the file containing the Chemkin-compatible input.
FILE2	Name of the file containing the chemistry thermodynamics database.
FILE3	Name of the file containing the chemistry transport properties database.

*CHEMISTRY_PATH

Purpose: To specify one or more search paths to look for chemistry database files.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	DIR							
Type	A							

VARIABLE	DESCRIPTION
DIR	Directory path to add to the search set.

*DUALCESE

The keyword *DUALCESE provides input data for the dual Conservation Element/Solution Element (dual CESE) compressible fluid solver:

DUALCESE_BOUNDARY_AXISYMMETRIC_*{OPTION}

DUALCESE_BOUNDARY_CYCLIC_*{OPTION}

DUALCESE_BOUNDARY_FSI_*{OPTION}

DUALCESE_BOUNDARY_NON_REFLECTIVE_*{OPTION}

DUALCESE_BOUNDARY_PRESCRIBED_*{OPTION}

DUALCESE_BOUNDARY_PRESCRIBED_HYBRID_*{OPTION}

DUALCESE_BOUNDARY_PRESCRIBED_PHASE_CHANGE_*{OPTION}

DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE_*{OPTION}

DUALCESE_BOUNDARY_REFLECTIVE_*{OPTION}

DUALCESE_BOUNDARY_SLIDING_*{OPTION}

DUALCESE_BOUNDARY_SOLID_WALL_*{OPTION1}_{OPTION2}

*DUALCESE_CONTROL_LIMITER

*DUALCESE_CONTROL_MESH_MOV

*DUALCESE_CONTROL_SOLVER (deprecated as of R14)

*DUALCESE_CONTROL_TIMESTEP

*DUALCESE_DATABASE_HISTORY_ELEMENT_SET

*DUALCESE_DATABASE_HISTORY_GLOBALS

*DUALCESE_DATABASE_HISTORY_NODE_SET

*DUALCESE_DATABASE_HISTORY_POINT_SET

*DUALCESE_DATABASE_HISTORY_SEGMENT_SET

*DUALCESE_D3PLOT

*DUALCESE_D3PLOT_FLUID_SSID

*DUALCESE

*DUALCESE_ELE2D

*DUALCESE_ELE3D

*DUALCESE_ELEMENTSET

*DUALCESE_EOS_CAV_HOMOG_EQUILIB

*DUALCESE_EOS_COCHRAN_CHAN

*DUALCESE_EOS_COOLPROP

*DUALCESE_EOS_IDEAL_GAS

*DUALCESE_EOS_INFLATOR1

*DUALCESE_EOS_INFLATOR2

*DUALCESE_EOS_JWL

*DUALCESE_EOS_NASG

*DUALCESE_EOS_REFPROP

*DUALCESE_EOS_REFPROP_PATH

*DUALCESE_EOS_SET (deprecated as of R14)

*DUALCESE_EOS_STIFFENED_GAS

*DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED

*DUALCESE_FSI_EXCLUDE

*DUALCESE_INCLUDE_MODEL

*DUALCESE_INITIAL

DUALCESE_INITIAL_*{OPTION}

*DUALCESE_INITIAL_HYBRID

*DUALCESE_INITIAL_HYBRID_SET

*DUALCESE_INITIAL_PHASE_CHANGE

*DUALCESE_INITIAL_PHASE_CHANGE_SET

*DUALCESE_INITIAL_TWO-PHASE

*DUALCESE_INITIAL_TWO-PHASE_SET

*DUALCESE_MAT_GAS

*DUALCESE_MAT_GAS_0

*DUALCESE_MAT_GAS_2

*DUALCESE_MESH_GEOMETRY

*DUALCESE_MESH_PART

*DUALCESE_MODEL

*DUALCESE_NODE2D

*DUALCESE_NODE3D

*DUALCESE_NODESET

*DUALCESE_PART (deprecated as of R14)

*DUALCESE_PART_MULTIPHASE (deprecated as of R14)

*DUALCESE_POINTSET

*DUALCESE_REACTION_RATE_IG

*DUALCESE_REACTION_RATE_IG_REDUCED

*DUALCESE_REACTION_RATE_P_DEPEND

*DUALCESE_SEGMENTSET

*DUALCESE_SOLVER_SELECTION

*DUALCESE_SOLVER_CAV_EQNS

*DUALCESE_SOLVER_EULER_EQNS

*DUALCESE_SOLVER_HYBRID_MULTIPHASE

*DUALCESE_SOLVER_NAVIER_STOKES

*DUALCESE_SOLVER_PHASE_CHANGE

*DUALCESE_SOLVER_TWO-PHASE_MULTIPHASE

An additional keyword option TITLE may be appended to the *DUALCESE keywords. If this option is used, then an addition line is read for the DUALCESE card in 80a format which can be used to describe that particular DUALCESE card. At present, the title serves no purpose other than to perhaps lend clarity to input decks.

Dual CESE Keyword Deck Structure:

The structure of the keyword setup for the *DUALCESE solvers is different from the way most keyword input is handled in the LS-DYNA input phase. This stems from the fact that there can be several *DUALCESE models in the same problem. In order to deal with this, each such model is restricted to be specified with one file hierarchy that starts with the keyword file designated with the *DUALCESE_MODEL card.

That keyword file can include any number of other keyword files with the *DUALCESE_-INCLUDE_MODEL card, and each of those files can in turn include other keyword files, again with the *DUALCESE_INCLUDE_MODEL card. Standard *INCLUDE cards are not allowed. In fact, in each file in the file hierarchy of a *DUALCESE_MODEL card, only *DUALCESE cards may be used. The only exception to this is when the fluid mesh is defined using *MESH cards; a fatal error will be encountered when using other non-*DUALCESE keywords. Any required non-*DUALCESE keyword cards should be defined in some other place in the keyword input (outside the scope of the *DUALCESE_MODEL keyword card).

The mesh for each dual CESE model must be defined within the keyword input file hierarchy for that model. If a mesh created with *MESH cards is used, the *MESH cards for that mesh must all be defined within the scope of the *DUALCESE_MODEL card.

With R14, we introduce an updated, preferred way of specifying the dual CESE problem. The required or recommended keywords to use are:

*DUALCESE_MESH_GEOMETRY

*DUALCESE_MESH_PART

*DUALCESE_SOLVER_SELECTION

along with one of the following specific solver cards:

*DUALCESE_SOLVER_CAV_EQNS

*DUALCESE_SOLVER_EULER_EQNS

*DUALCESE_SOLVER_HYBRID_MULTIPHASE

*DUALCESE_SOLVER_NAVIER_STOKES

*DUALCESE_SOLVER_PHASE_CHANGE (added in R15)

*DUALCESE_SOLVER_TWO-PHASE_MULTIPHASE

The following keywords are deprecated as of R14:

*DUALCESE_CONTROL_SOLVER

*DUALCESE_EOS_SET

*DUALCESE_PART

*DUALCESE_PART_MULTIPHASE

We also added time history cards with R14 that cause certain output to the binout file:

*DUALCESE_DATABASE_HISTORY_ELEMENT_SET

*DUALCESE_DATABASE_HISTORY_GLOBALS

*DUALCESE_DATABASE_HISTORY_NODE_SET

*DUALCESE_DATABASE_HISTORY_POINT_SET

*DUALCESE_DATABASE_HISTORY_SEGMENT_SET

We support two external equation-of-state (EOS) libraries, REFPROP and COOLPROP, for dual CESE. Including these libraries is complex, so we will clarify how to include them here. Each of them is accessed through a shared library that has to be loaded into LS-DYNA at runtime with *MODULE_LOAD like in the follwing:

```
*MODULE_LOAD
UserA                   DUALCESE REFPROP
<path to the installed REFPROP shared library>
```

Note that this *MODULE_LOAD card must not be given inside a keyword file in the file hierarchy of a *DUALCESE_MODEL card. As discussed above, this is the case for all non-*DUALCESE keyword cards. Note also that since *MODULE is not available in the Windows version of LS-DYNA, this capability cannot be used in that version.

REFPROP and COOLPROP Libraries:

While the REFPROP v10.0 version library and its directory of data sets is provided by ANSYS, COOLPROP libraries are not provided by ANSYS.

For the COOLPROP shared library, you can find the current production version here:

https://sourceforge.net/projects/coolprop/files/CoolProp/6.3.0/shared_library/Linux/64bit/

Multiphase Capabilities:

The multiphase capabilities added for R13 with modifications in R14 all have FSI capabilities, including with structural element failure. This includes the 'hybrid' multiphase,

'two-phase' multiphase, and the cavitation solvers. These keywords are involved with these capabilities:

*DUALCESE_BOUNDARY_PRESCRIBED_HYBRID

*DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE

*DUALCESE_EOS_CAV_HOMOG_EQUILIB

*DUALCESE_EOS_COCHRAN_CHAN

*DUALCESE_EOS_JWL

*DUALCESE_EOS_SET (deprecated as of R14; use one of the *DUALCESE_-SOLVER_... cards instead)

*DUALCESE_EOS_STIFFENED_GAS

*DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED

*DUALCESE_INITIAL_HYBRID

*DUALCESE_INITIAL_HYBRID_SET

*DUALCESE_INITIAL_TWO-PHASE

*DUALCESE_INITIAL_TWO-PHASE_SET

*DUALCESE_PART_MULTIPHASE (deprecated as of R14; use *DUALCESE_-MESH_PART instead)

*DUALCESE_REACTION_RATE_IG

*DUALCESE_REACTION_RATE_IG_REDUCED

*DUALCESE_REACTION_RATE_P_DEPEND

*DUALCESE_SOLVER_HYBRID_MULTIPHASE (added in R14)

*DUALCESE_SOLVER_TWO-PHASE_MULTIPHASE (added in R14)

Comparison to CESE:

The capabilities implemented in the dual CESE solvers are only a part of what is available in the *CESE solvers, especially those that involve couplings with the *CHEMISTRY and *STOCHASTIC_PARTICLE solvers. We plan to port some of those capabilities to the *DUALCESE solvers as well.

DUALCESE_BOUNDARY_AXISYMMETRIC_*OPTION

Available options are:

MSURF

SEGMENT_SET

Purpose: Define an axisymmetric boundary condition on the axisymmetric axis for the 2D axisymmetric dual CESE compressible flow solver.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Summary:

Card 1a. This card is included for the MSURF keyword option. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

MSPID							

Card 1b. This card is included for the SEGMENT_SET keyword option. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

SSID							

Data Card Definitions:

Surface Part Card. Card 1 used when the MSURF keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards

Set Card. Card 1 used when the SEGMENT_SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
SSID	Segment set ID for the segment set created with *DUALCESE_SEGMENTSET

Remarks:

This boundary condition can only be used on the axisymmetric axis for the 2D axisymmetric dual CESE fluid solver.

DUALCESE_BOUNDARY_CYCLIC_*OPTION

Available options are:

MSURF

SEGMENT_SET

Purpose: Define a cyclic (periodic) boundary condition for dual CESE compressible flows. This cyclic boundary condition can be used on periodic boundary surfaces.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Summary:

Card Sets. The following sequence of cards comprises a *single set*. LS-DYNA will continue reading these cards sets until the next keyword ("*") card is encountered.

Card 1a. This card is included if the MSURF keyword option is used.

MSPID1	MSPID2	CYCTYP					

Card 1b. This card is included if the SEGMENT_SET keyword option is used.

SSID1	SSID2	CYCTYP					

Card 2a. This card is included when CYCTYP = 1.

AXISX1	AXISY1	AXISZ1	DIRX	DIRY	DIRZ	ROTANG	

Card 2b. This card is included when CYCTYP = 2.

TRANSX	TRANSY	TRANSZ					

Data Card Definitions:

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID1	MSPID2	CYCTYP					
Type	I	I	I					
Default	none	none	0					
Remarks			1, 2					

VARIABLE	DESCRIPTION
MSPID1, MSPID2	Mesh surface part IDs that are referenced by *MESH_SURFACE_ELEMENT cards
CYCTYP	Relationship between the two cyclic boundary condition surfaces: EQ.0: Relationship determined by LS-DYNA (default) EQ.1: The first surface is rotated about an axis to match the second surface. EQ.2: The faces of the first surface are translated in a given direction to obtain the corresponding faces on the second surface.

Set Card. Card 1 format used when the SEGMENT_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID1	SSID2	CYCTYP					
Type	I	I	I					
Default	none	none	0					
Remarks			1, 3					

VARIABLE	DESCRIPTION
SSID1, SSID2	Segment set IDs for the segment sets created with *DUALCESE_-SEGMENTSET
CYCTYP	Relationship between the two cyclic boundary condition surfaces: EQ.0: Relationship determined by LS-DYNA (default) EQ.1: The first surface is rotated about an axis to match the second surface. EQ.2: The faces of the first surface are translated in a given direction to obtain the corresponding faces on the second surface.

Rotation Case Card. Additional card when CYCTYP = 1.

Card 2a	1	2	3	4	5	6	7	8
Variable	AXISX1	AXISY1	AXISZ1	DIRX	DIRY	DIRZ	ROTANG	
Type	F	F	F	F	F	F	F	
Default	0.0	0.0	0.0	none	none	none	none	

VARIABLE	DESCRIPTION
AXIS[X,Y,Z]1	A point on the axis of rotation for the transformation between the surfaces
DIR[X,Y,Z]	The direction which together with AXIS[X,Y,Z]1 defines the axis of rotation for the transformation between the surfaces
ROTANG	The angle of rotation (in degrees) that transforms the centroid of each face on the first surface to the centroid of the corresponding face on the second surface

Translation Case Card. Additional card when CYCTYP = 2.

Card 2b	1	2	3	4	5	6	7	8
Variable	TRANSX	TRANSY	TRANSZ					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
TRANS[X,Y,Z]	The translation direction that enables the identification of the segment in the second surface that matches a segment in the first surface

Remarks:

1. **Unspecified Relationship between Surfaces.** For the case CYCTYP = 0, LS-DYNA examines the geometry of two faces of the two surfaces in order to determine if the surfaces are approximately parallel (CYCTYP = 2) or related through a rotation (CYCTYP = 1). The geometric parameters required are then computed.

2. **MSURF.** For the MSURF option, each mesh surface part must contain the same number of mesh surface elements. The mesh surface elements in each mesh surface part are internally ordered for pairwise matching between the two mesh surface parts.

3. **SEGMENT_SET.** For the SEGMENT_SET option, each segment set must contain the same number segments. The segments in each set are internally ordered for pairwise matching between the two sets.

DUALCESE_BOUNDARY_FSI_*OPTION

Available options are:

MSURF

SEGMENT_SET

Purpose: Define an FSI boundary condition for the moving mesh dual CESE compressible flow solver. This keyword must not be combined with the dual CESE immersed-boundary method FSI solver in the same dual CESE part on the same dual CESE mesh. Doing so will result in an error termination condition.

This boundary condition must be applied on a surface of the dual CESE computational domain that is co-located with surfaces of the outside boundary of the structural mesh. The nodes of the two meshes will generally not be shared.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Summary:

Card 1a. This card is included for the MSURF keyword option. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

MSPID	REF_P						

Card 1b. This card is included for the SEGMENT_SET keyword option. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

SSID	REF_P						

Data Card Definitions:

Surface Part Card. Card 1 used when the MSURF keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	REF_P						
Type	I	F						
Default	none	0.0						

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
REF_P	Ambient/reference pressure of the fluid domain on the side opposite this structural interface to the fluid simulation domain. This ambient pressure only needs to be specified in the case where the FSI structural part(s) connected with this FSI interface are not immersed in the dual CESE mesh. This reference pressure defaults to 0.0 since moving mesh FSI calculations most often involve structures surrounded by the dual CESE mesh, and there is no need for a reference pressure in that case.

Set Card. Card 1 used when the SEGMENT_SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID	REF_P						
Type	I	F						
Default	none	0.0						

VARIABLE	DESCRIPTION
SSID	Segment set ID for the segment set created with *DUALCESE_SEGMENTSET

VARIABLE	DESCRIPTION
REF_P	Ambient/reference pressure of the fluid domain on the side opposite this structural interface to the fluid simulation domain. This ambient pressure only needs to be specified in the case where the FSI structural part(s) connected with this FSI interface are not immersed in the dual CESE mesh. This reference pressure defaults to 0.0 since moving mesh FSI calculations most often involve structures surrounded by the dual CESE mesh, and there is no need for a reference pressure in that case..

Remarks:

This boundary condition card is also needed for conjugate heat transfer problems with the moving mesh dual CESE solver. But the conjugate heat transfer capability is not yet implemented in the dual CESE solver.

DUALCESE_BOUNDARY_NON_REFLECTIVE_*OPTION

Available options are:

MSURF

SEGMENT_SET

Purpose: Define a passive boundary condition for dual CESE compressible flows. This non-reflective boundary condition provides an artificial computational boundary for an open boundary that is passive.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards are used to specify the dual CESE mesh.

Card Summary:

Card 1a. This card is included when the MSURF keyword option is used. Include as many cards as necessary. This input ends at the next keyword ("*") card.

MSPID	DIRX	DIRY	DIRZ				

Card 1b. This card is included when the SEGMENT_SET keyword option is used. Include as many cards as necessary. This input ends at the next keyword ("*") card.

SSID	DIRX	DIRY	DIRZ				

Data Card Definitions:

Surface Part Card. Card 1 used when the MSURF keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	DIRX	DIRY	DIRZ				
Type	I	F	F	F				
Default	none	0.0	0.0	0.0				

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
DIRX, DIRY, DIRZ	If this vector is nonzero, then it is used as the prescribed flow direction.

Set Card. Card 1 used when the SET keyword option is active. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID	DIRX	DIRY	DIRZ				
Type	I	F	F	F				
Default	none	0.0	0.0	0.0				

VARIABLE	DESCRIPTION
SSID	Segment set ID for the segment set created with *DUALCESE_SEGMENTSET
DIRX, DIRY, DIRZ	If this vector is nonzero, then it is used as the prescribed flow direction.

Remarks:

1. **Boundary Surface Flow.** This boundary condition is usually imposed on an open surface that is far from the main disturbed flow (the further away, the better), meaning the flow on the boundary surface should be almost uniform.

2. **Default Boundary Condition**. If any boundary segment has not been assigned a boundary condition by any of the *DUALCESE_BOUNDARY_… cards, then it will automatically be assigned this non-reflective boundary condition.

DUALCESE_BOUNDARY_PRESCRIBED_*OPTION

Available options include:

MSURF

SEGMENT_SET

Purpose: For the dual CESE compressible flow solver, set boundary values for velocity, density, pressure and temperature. Boundary values are applied at the centroid of elements connected with this boundary.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET card should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.
2. Card 2 reads in load curve IDs.
3. Card 3 reads in scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Card Summary:

Card 1a. This card is included if the MSURF keyword option is used.

MSPID	IDCOMP	DIRX	DIRY	DIRZ			

Card 1b. This card is included if the SEGMENT_SET keyword option is used.

SSID	IDCOMP	DIRX	DIRY	DIRZ			

Card 2. This card is required.

LC_U	LC_V	LC_W	LC_RHO	LC_P	LC_T		

Card 3. This card is required.

SF_U	SF_V	SF_W	SF_RHO	SF_P	SF_T		

Data Card Definitions:

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	IDCOMP	DIRX	DIRY	DIRZ			
Type	I	I	F	F	F			
Default	none	none	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].
DIRX, DIRY, DIRZ	If this vector is non-zero, then it is used as the prescribed flow direction.

Set Card. Card 1 format used when the SEGMENT_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID	IDCOMP	DIRX	DIRY	DIRZ			
Type	I	I	F	F	F			
Default	none	none	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
SSID	ID for the segment set created with *DUALCESE_SEGMENTSET
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].
DIRX, DIRY, DIRZ	If this vector is non-zero, then it is used as the prescribed flow direction.

Load Curve Card.

Card 2	1	2	3	4	5	6	7	8
Variable	LC_U	LC_V	LC_W	LC_RHO	LC_P	LC_T		
Type	I	I	I	I	I	I		
Remarks	1	1	1	1	1	1		

VARIABLE	DESCRIPTION
LC_U	Load curve ID (see *DEFINE_CURVE) to describe the x-component of the velocity as a function of time or function ID (see *DEFINE_FUNCTION) to give the x-component of the velocity as a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time). EQ.0: x-component of velocity is a constant with value SF_U. EQ.-1: x-component of velocity is computed by the solver.
LC_V	Load curve ID to describe the y-component of the velocity as a function of time or function ID to give the y-component of the velocity as a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time). EQ.0: y-component of velocity is a constant with value SF_V. EQ.-1: y-component of velocity is computed by the solver.
LC_W	Load curve ID to describe the z-component of the velocity as a function of time or function ID to give the z-component of the

VARIABLE	DESCRIPTION
	velocity as a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time). EQ.0: z-component of velocity is a constant with value SF_W. EQ.-1: z-component of velocity is computed by the solver.
LC_RHO	Load curve ID to describe the density as a function of time or function ID to give the density as a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time). EQ.0: Density is a constant with value SF_RHO. EQ.-1: Density is computed by the solver.
LC_P	Load curve ID to describe the pressure as a function of time or function ID to give the pressure as a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time). EQ.0: Pressure is a constant with value SF_P. EQ.-1: Pressure is computed by the solver.
LC_T	Load curve ID to describe the temperature as a function of time or function ID to give the temperature as a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time). EQ.0: Temperature is a constant with value SF_T. EQ.-1: Temperature is computed by the solver.

Scale Factor Card.

Card 3	1	2	3	4	5	6	7	8
Variable	SF_U	SF_V	SF_W	SF_RHO	SF_P	SF_T		
Type	F	F	F	F	F	F		
Default	1.0	1.0	1.0	1.0	1.0	1.0		

VARIABLE	DESCRIPTION
SF_U	Scale factor for LC_U

VARIABLE	DESCRIPTION
SF_V	Scale factor for LC_V
SF_W	Scale factor for LC_W
SF_RHO	Scale factor for LC_RHO
SF_P	Scale factor for LC_P
SF_T	Scale factor for LC_T

Remarks:

1. **Consistent Boundary Values.** On each centroid or set of centroids, the variables (v_x, v_y, v_z, ρ, P, T) that are given values must be consistent and make the model well-posed (meaning be such that the solution of the model exists, is unique, and is physical).

DUALCESE_BOUNDARY_PRESCRIBED_HYBRID_*OPTION

Available options include:

MSURF

SEGMENT_SET

Purpose: For the dual CESE compressible flow solver, set boundary values for velocity, density, pressure, temperature, and other values in the hybrid multiphase model. Boundary values are applied at the centroid of elements connected with this boundary.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET card should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.
2. Cards 2 and 3 provide load curve IDs.
3. Cards 4 and 5 provide scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Card Summary:

Card 1a. This card is included if the keyword option is set to MSURF.

MSPID	IDCOMP	DIRX	DIRY	DIRZ			

Card 1b. This card is included if the keyword option is set to SSID.

SSID	IDCOMP	DIRX	DIRY	DIRZ			

Card 2. This card is required.

LC_Z1	LC_RA	LC_U	LC_V	LC_W	LC_D1	LC_DA	LC_DB

Card 3. This card is required.

LC_P	LC_T						

Card 4. This card is required.

SF_Z1	SF_RA	SF_U	SF_V	SF_W	SF_D1	SF_DA	SF_DB

Card 5. This card is required.

SF_P	SF_T						

Data Card Definitions:

Surface Part Set Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	IDCOMP	DIRX	DIRY	DIRZ			
Type	I	I	F	F	F			
Default	none	none	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].
DIRX, DIRY, DIRZ	If this vector is nonzero, then it is used as the prescribed flow direction.

Segment Set Card. Card 1 format used when the SEGMENT_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID	IDCOMP	DIRX	DIRY	DIRZ			
Type	I	I	F	F	F			
Default	none	none	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
SSID	ID for the segment set created with *DUALCESE_SEGMENTSET
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].
DIRX, DIRY, DIRZ	If this vector is nonzero, then it is used as the prescribed flow direction.

Load Curve Card. See Remark 1.

Card 2	1	2	3	4	5	6	7	8
Variable	LC_Z1	LC_RA	LC_U	LC_V	LC_W	LC_D1	LC_DA	LC_DB
Type	I	I	I	I	I	I		

VARIABLE	DESCRIPTION
LC_Z1	Load curve ID or function ID to describe the volume fraction of material 1 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The volume fraction is a constant with value SF_Z1. EQ.-1: The volume fraction is computed by the solver.
LC_RA	Load curve or function ID to describe the mass fraction of reactant (material α) with respect to the explosive mixture (material 2) as a

VARIABLE	DESCRIPTION
	function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The mass fraction is a constant with value SF_RA. EQ.-1: The mass fraction is computed by the solver.
LC_U	Load curve or defined function ID to describe the x-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The x-component of velocity is a constant with value SF_U. EQ.-1: The x-component of velocity is computed by the solver.
LC_V	Load curve or defined function ID to describe the y-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The y-component of velocity is a constant with value SF_V. EQ.-1: The y-component of velocity is computed by the solver.
LC_W	Load curve or defined function ID to describe the z-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The z-component of velocity is a constant with value SF_W. EQ.-1: The z-component of velocity is computed by the solver.
LC_D1	Load curve or defined function ID to describe the density of the first multiphase material as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The density of the first multiphase material is a constant with value SF_D1. EQ.-1: The density of the first multiphase material is computed by the solver.

VARIABLE	DESCRIPTION
LC_DA	Load curve or defined function ID to describe the density of the reactant (material α) as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The density of the reactant is a constant with value SF_DA. EQ.-1: The density of the reactant is computed by the solver.
LC_DB	Load curve or defined function ID to describe the density of the product (material β) as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The density of the product is a constant with value SF_DB. EQ.-1: The density of the product is computed by the solver.

Load Curve Card 2. See Remark 1.

Card 3	1	2	3	4	5	6	7	8
Variable	LC_P	LC_T						
Type	I	I						

VARIABLE	DESCRIPTION
LC_P	Load curve or defined function ID to describe the pressure as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The pressure is a constant with value SF_P. EQ.-1: The pressure is computed by the solver.
LC_T	Load curve or defined function ID to describe the temperature as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The temperature is a constant with value SF_T.

VARIABLE	DESCRIPTION
	EQ.-1: The temperature is computed by the solver.

Scale Factor Card.

Card 4	1	2	3	4	5	6	7	8
Variable	SF_Z1	SF_RA	SF_U	SF_V	SF_W	SF_D1	SF_DA	SF_DB
Type	F	F	F	F	F	F	F	F
Default	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

VARIABLE	DESCRIPTION
SF_Z	Scale factor for LC_Z1
SF_RA	Scale factor for LC_RA
SF_U	Scale factor for LC_U
SF_V	Scale factor for LC_V
SF_W	Scale factor for LC_W
SF_D1	Scale factor for LC_D1
SF_DA	Scale factor for LC_DA
SF_DB	Scale factor for LC_DB

Scale Factor Card 2.

Card 5	1	2	3	4	5	6	7	8
Variable	SF_P	SF_T						
Type	F	F						
Default	1.0	1.0						

VARIABLE	DESCRIPTION
SF_P	Scale factor for LC_P
SF_T	Scale factor for LC_T

Remark:

1. **Consistent Boundary Values.** On each centroid or set of element centroids, the variables ($v_x, v_y, v_z, \rho, P, T, ...$) that are given values must be consistent and make the model well-posed (meaning be such that the solution of the model exists, is unique, and is physical).

DUALCESE_BOUNDARY_PRESCRIBED_PHASE_CHANGE_*OPTION

Available options are:

MSURF

SEGMENT_SET

Purpose: For the dual CESE compressible flow solver, set boundary values for velocity, density, pressure, temperature, and other values in the phase-change multiphase model. Boundary values are applied at the centroid of outside elements connected with this boundary.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option card should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.
2. Cards 2 and 3 provide load curve IDs.
3. Cards 4 and 5 provide scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Card Summary:

Card 1a. This card is included for the MSURF keyword option.

MSPID		DIRX	DIRY	DIRZ			

Card 1b. This card is included for the SEGMENT_SET keyword option.

SSID		DIRX	DIRY	DIRZ			

Card 2. This card is required.

LCIDU	LCIDV	LCIDW	LCIDRHO	LCIDP	LCIDT	LCIDY1	LCIDY2

Card 3. This card is required.

LCIDY3							

Card 4. This card is required.

SFU	SFV	SFW	SFRHO	SFP	SFT	SFY1	SFY2

Card 5. This card is required.

SFY3							

Data Card Definitions:

Surface Part Card. Card 1 used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID		DIRX	DIRY	DIRZ			
Type	I		F	F	F			
Default	none		0.0	0.0	0.0			

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
DIRX, DIRY, DIRZ	If nonzero, vector giving the prescribed flow direction

Set Card. Card 1 used when the SEGMENT_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID		DIRX	DIRY	DIRZ			
Type	I		F	F	F			
Default	none		0.0	0.0	0.0			

VARIABLE	DESCRIPTION
SSID	Segment set ID for the segment set created with *DUALCESE_-SEGMENTSET
DIRX, DIRY, DIRZ	If nonzero, vector giving the prescribed flow direction

Card 2	1	2	3	4	5	6	7	8
Variable	LCIDU	LCIDV	LCIDW	LCIDRHO	LCIDP	LCIDT	LCIDY1	LCIDY2
Type	I	I	I	I	I	I	I	I
Default	0	0	0	0	0	0	0	0
Remarks	1	1	1	1	1	1	1	1

VARIABLE	DESCRIPTION
LCIDU	Load curve or defined function ID to describe the *xx*-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The *x*-component of velocity is a constant with value SFU. EQ.-1: The *x*-component of velocity is computed by the solver.
LCIDV	Load curve or defined function ID to describe the *yy*-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The *y*-component of velocity is a constant with value SFV. EQ.-1: The *y*-component of velocity is computed by the solver.
LCIDW	Load curve or defined function ID to describe the *zz*-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.

VARIABLE	DESCRIPTION
	EQ.0: The z-component of velocity is a constant with value SFW.
	EQ.-1: The z-component of velocity is computed by the solver.
LCIDRHO	Load curve or defined function ID to describe the mixture density as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.
	EQ.0: The density is a constant with value SFRHO.
	EQ.-1: The density is computed by the solver.
LCIDP	Load curve or defined function ID to describe the pressure as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.
	EQ.0: The pressure is a constant with value SFP.
	EQ.-1: The pressure is computed by the solver.
LCIDT	Load curve or defined function ID to describe the temperature as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.
	EQ.0: The temperature is a constant with value SFT.
	EQ.-1: The temperature is computed by the solver.
LCIDY1	Load curve or defined function ID to describe the mass fraction of material 1 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.
	EQ.0: The mass fraction is a constant with value SFY1.
	EQ.-1: The mass fraction is computed by the solver.
LCIDY2	Load curve or defined function ID to describe the mass fraction of material 2 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.
	EQ.0: The mass fraction is a constant with value SFY2.
	EQ.-1: The mass fraction is computed by the solver.

Card 3	1	2	3	4	5	6	7	8
Variable	LCIDY3							
Type	I							
Default	0							
Remarks	1							

VARIABLE	DESCRIPTION
LCIDY3	Load curve or defined function ID to describe the mass fraction of material 3 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The mass fraction is a constant with value SFY3. EQ.-1: The mass fraction is computed by the solver.

Card 4	1	2	3	4	5	6	7	8
Variable	SFU	SFV	SFW	SFRH0	SFP	SFT	SFY1	SFY2
Type	F	F	F	F	F	F	F	F
Default	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

VARIABLE	DESCRIPTION
SFU	Scale factor for LCIDU
SFV	Scale factor for LCIDV
SFW	Scale factor for LCIDW
SFRHO	Scale factor for LCIDRHO
SFP	Scale factor for LCIDP

VARIABLE	DESCRIPTION
SFT	Scale factor for LCIDT
SFY1	Scale factor for LCIDY1
SFY2	Scale factor for LCIDY2

Card 5	1	2	3	4	5	6	7	8
Variable	SFY3							
Type	F							
Default	1.0							

VARIABLE	DESCRIPTION
SFY3	Scale factor for LCIDY3

Remarks:

1. **Consistent Boundary Values.** On each centroid or set of element centroids, the variables $(v_x, v_y, v_z, \rho, P, T, \ldots)$ that are given values must be consistent and make the model well-posed (meaning be such that the solution of the model exists, is unique, and is physical).

DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE_*OPTION

Available options include:

MSURF

SEGMENT_SET

Purpose: For the dual CESE compressible flow solver, set boundary values for velocity, density, pressure, temperature, and other values in the two-phase multiphase model. Boundary values are applied at the centroid of elements connected with this boundary. OPTION = SEGMENT_SET is for user defined meshes whereas OPTION = MSURF is associated with the automatic volume mesher (See *MESH keywords).

That is, the MSURF option is used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET card is used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards are used to define the dual CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.
2. Card 2 provides load curve IDs.
3. Card 3 provides scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Card Summary:

Card 1a. This card is included if the keyword option is set to MSURF.

MSPID	IDCOMP	DIRX	DIRY	DIRZ			

Card 1b. This card is included if the keyword option is set to SSID.

SSID	IDCOMP	DIRX	DIRY	DIRZ			

Card 2. This card is required.

LC_Z1	LC_U	LC_V	LC_W	LC_D1	LC_D2	LC_P	LC_T

Card 3. This card is required.

SF_Z1	SF_U	SF_V	SF_W	SF_D1	SF_D2	SF_P	SF_T

Data Card Definitions:

Surface Part Set Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	IDCOMP	DIRX	DIRY	DIRZ			
Type	I	I	F	F	F			
Default	none	none	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].
DIRX, DIRY, DIRZ	If this vector is nonzero, then it is used as the prescribed flow direction.

Segment Set Card. Card 1 format used when the SEGMENT_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID	IDCOMP	DIRX	DIRY	DIRZ			
Type	I	I	F	F	F			
Default	none	none	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
SSID	ID for the segment set created with *DUALCESE_SEGMENTSET
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].
DIRX, DIRY, DIRZ	If this vector is nonzero, then it is used as the prescribed flow direction.

Load Curve Card. See Remark 1.

Card 2	1	2	3	4	5	6	7	8
Variable	LC_Z1	LC_U	LC_V	LC_W	LC_D1	LC_D2	LC_P	LC_T
Type	I	I	I	I	I	I	I	I

VARIABLE	DESCRIPTION
LC_Z1	Load curve or defined function ID to describe the volume fraction of material 1 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The volume fraction is a constant with value SF_Z1. EQ.-1: The volume fraction is computed by the solver.
LC_U	Load curve or defined function ID to describe the x-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The x-component of velocity is a constant with value SF_U. EQ.-1: The x-component of velocity is computed by the solver.
LC_V	Load curve or defined function ID to describe the y-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively.

VARIABLE	DESCRIPTION
	EQ.0: The y-component of velocity is a constant with value SF_V. EQ.-1: The y-component of velocity is computed by the solver.
LC_W	Load curve or defined function ID to describe the z-component of the velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The z-component of velocity is a constant with value SF_W. EQ.-1: The z-component of velocity is computed by the solver.
LC_D1	Load curve or defined function ID to describe the density of material 1 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The density is a constant with value SF_D1. EQ.-1: The density is computed by the solver.
LC_D2	Load curve or defined function ID to describe the density of material 2 as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The density is a constant with value SF_D2. EQ.-1: The density is computed by the solver.
LC_P	Load curve or defined function ID to describe the pressure as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The pressure is a constant with value SF_P. EQ.-1: The pressure is computed by the solver.
LC_T	Load curve or defined function ID to describe the temperature as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The temperature is a constant with value SF_T.

VARIABLE	DESCRIPTION
	EQ.-1: The temperature is computed by the solver.

Scale Factor Card.

Card 3	1	2	3	4	5	6	7	8
Variable	SF_Z1	SF_U	SF_V	SF_W	SF_D1	SF_D2	SF_P	SF_T
Type	F	F	F	F	F	F	F	F
Default	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

VARIABLE	DESCRIPTION
SF_Z	Scale factor for LC_Z1
SF_U	Scale factor for LC_U
SF_V	Scale factor for LC_V
SF_W	Scale factor for LC_W
SF_D1	Scale factor for LC_D1
SF_D2	Scale factor for LC_D2
SF_P	Scale factor for LC_P
SF_T	Scale factor for LC_T

Remark:

1. **Consistent Boundary Values.** On each centroid or set of element centroids, the variables ($v_x, v_y, v_z, \rho, P, T, \ldots$) that are given values must be consistent and make the model well-posed (meaning be such that the solution of the model exists, is unique, and is physical).

DUALCESE_BOUNDARY_PRESCRIBED_VN_*OPTION

Available options include:

MSURF

SEGMENT_SET

Purpose: For the dual CESE compressible flow solver, set boundary values for the normal velocity, density, pressure and temperature. Boundary values are applied at the centroid of elements connected with this boundary. This boundary condition differs from *DUALCESE_BOUNDARY_PRESCRIBED_VN in that the normal velocity is prescribed instead of each velocity component.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Sets:

A set of data cards for this keyword consists of 3 of the following cards:

1. Card 1 specifies the object to which the boundary condition is applied. Its format depends on the keyword option.

2. Card 2 reads in load curve IDs.

3. Card 3 reads in scale factors.

For each boundary condition to be specified include one set of cards. This input ends at the next keyword ("*") card.

Card Summary:

Card 1a. This card is included if the MSURF keyword option is used.

MSPID	IDCOMP						

Card 1b. This card is included if the SEGMENT_SET keyword option is used.

SSID	IDCOMP						

Card 2. This card is required.

LC_VN			LC_RHO	LC_P	LC_T		

Card 3. This card is required.

SF_VN			SF_RHO	SF_P	SF_T		

Data Card Definitions:

Surface Part Card. Card 1 format used when the MSURF keyword option is active.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	IDCOMP						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].

Set Card. Card 1 format used when the SEGMENT_SET keyword option is active.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID	IDCOMP						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
SSID	ID for the segment set created with *DUALCESE_SEGMENTSET

VARIABLE	DESCRIPTION
IDCOMP	For inflow boundaries in problems involving chemical reacting flows, the chemical mixture of the fluid entering the domain is defined with a *CHEMISTRY_COMPOSITION card with this ID [Not yet available].

Load Curve Card. See Remark 1.

Card 2	1	2	3	4	5	6	7	8
Variable	LC_VN			LC_RHO	LC_P	LC_T		
Type	I			I	I	I		

VARIABLE	DESCRIPTION
LC_VN	Load curve or function ID to describe the normal velocity as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The normal velocity is a constant with value SF_VN. EQ.-1: The normal velocity is computed by the solver.
LC_RHO	Load curve ID to describe the density as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The density is a constant with value SF_RHO. EQ.-1: The density is computed by the solver.
LC_P	Load curve ID to describe the pressure as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The pressure is a constant with value SF_P. EQ.-1: The pressure is computed by the solver.
LC_T	Load curve ID to describe the temperature as a function of time or a function of position, velocity, temperature, pressure, and time, f(x, y, z, vx, vy, vz, temp, pres, time), respectively. EQ.0: The temperature is a constant with value SF_T. EQ.-1: The temperature is computed by the solver.

Scale Factor Card.

Card 3	1	2	3	4	5	6	7	8
Variable	SF_VN			SF_RHO	SF_P	SF_T		
Type	F			F	F	F		
Default	1.0			1.0	1.0	1.0		

VARIABLE	DESCRIPTION
SF_VN	Scale factor for LC_VN
SF_RHO	Scale factor for LC_RHO
SF_P	Scale factor for LC_P
SF_T	Scale factor for LC_T

Remarks:

1. **Consistent Boundary Values.** On each centroid or set of centroids, the variables (V_N, ρ, P, T) that are given values must be consistent and make the model well-posed (meaning be such that the solution of the model exists, is unique, and is physical).

DUALCESE_BOUNDARY_REFLECTIVE_*OPTION

Available options are:

MSURF

SEGMENT_SET

Purpose: Define a reflective boundary condition for the dual CESE compressible flow solver. This boundary condition can be applied on a symmetrical surface or a solid wall of the computational domain.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh.

Card Summary:

Card 1a. This card is included for the MSURF keyword option. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

MSPID							

Card 1b. This card is included for the SEGMENT_SET keyword option. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

SSID							

Data Card Definitions:

Surface Part Card. Card 1 format used when the MSURF keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards

Set Card. Card 1 format used when the SEGMENT_SET keyword option is active. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1b	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
SSID	Segment set ID for the segment set created with *DUALCESE_SEGMENTSET

Remarks:

This boundary condition has the same effect as a solid wall boundary condition for inviscid flows.

DUALCESE_BOUNDARY_SOLID_WALL_*OPTION1_{OPTION2}

For *OPTION1* the choices are:

MSURF

SEGMENT_SET

For *OPTION2* the choices are:

<BLANK>

ROTATE

Purpose: Define a solid wall boundary condition for the dual CESE compressible flow solver. This boundary condition can be applied at a solid boundary that is the physical boundary for the flow field. For inviscid flow, this will be a slip boundary condition; while for viscous flows, it is a no-slip boundary condition.

The MSURF option should be used when the dual CESE mesh has been created using *MESH cards. The SEGMENT_SET option should be used when *DUALCESE_ELE2D or *DUALCESE_ELE3D cards specify the dual CESE mesh. The ROTATE keyword option allows the boundary condition to rotate around an axis with a variable speed given by a load curve.

Card Summary:

Card Sets. The following sequence of cards comprises a *single set*. LS-DYNA will continue reading data card sets until the next keyword ("*") card is encountered.

Card 1a. This card is included for the MSURF keyword option *without* the ROTATE keyword option.

MSPID	LCID	VX	VY	VZ			

Card 1b. This card is included for the MSURF keyword option *with* the ROTATE keyword option.

MSPID	LCID	XP	YP	ZP	NX	NY	NZ

Card 1c. This card is included for the SEGMENT_SET keyword option *without* the ROTATE keyword option.

SSID	LCID	VX	VY	VZ			

Card 1d. This card is included for the SEGMENT_SET keyword option *with* the ROTATE keyword option.

SSID	LCID	XP	YP	ZP	NX	NY	NZ

Data Card Definitions:

Surface Part Card without Rotation. Card 1 format used when the MSURF keyword option is active *without* the ROTATE keyword option

Card 1a	1	2	3	4	5	6	7	8
Variable	MSPID	LCID	VX	VY	VZ			
Type	I	I	F	F	F			
Default	none	0	0.0	0.0	0.0			
Remarks		2	2	2	2			

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
LCID	Load curve ID scales the velocity vector specified with (VX, VY, VZ) to give the solid wall boundary movement. If not defined, the solid wall boundary moves with a constant velocity vector specified by (VX, VY, VZ).
VX, VY, VZ	Velocity vector of the solid wall boundary condition: LCID.EQ.0: Constant velocity vector specified with VX, VY, and VZ. LCID.NE.0: VX, VY, and VZ give the velocity vector that is scaled by LCID.

Surface Part Card with Rotation. Card 1 format used when the MSURF keyword option is active *with* the ROTATE keyword option.

Card 1b	1	2	3	4	5	6	7	8
Variable	MSPID	LCID	XP	YP	ZP	NX	NY	NZ
Type	I	I	F	F	F	F	F	F
Default	none	0	0.0	0.0	0.0	0.0	0.0	0.0

VARIABLE	DESCRIPTION
MSPID	Mesh surface part ID that is referenced by *MESH_SURFACE_ELEMENT cards
LCID	Load curve ID for specifying the rotating speed frequency in Hz. This input is required.
XP, YP, ZP	Coordinates for a point on the axis of rotation
NX, NY, NZ	Unit vector for specifying the direction of the axis of rotation. This is not used for the 2D case.

Segment Set Card without Rotation. Card 1 format used when the SEGMENT_SET keyword option is active *without* the ROTATE keyword option

Card 1c	1	2	3	4	5	6	7	8
Variable	SSID	LCID	VX	VY	VZ			
Type	I	I	F	F	F			
Default	none	0	0.0	0.0	0.0			
Remarks		2	2	2	2			

VARIABLE	DESCRIPTION
SSID	ID of the segment set created with *DUALCESE_SEGMENTSET

VARIABLE	DESCRIPTION
LCID	Load curve ID scales the velocity vector specified with (VX, VY, VZ) to give the solid wall boundary movement. If not defined, the solid wall boundary moves with a constant velocity vector specified by (VX, VY, VZ).
VX, VY, VZ	Velocity vector of the solid wall boundary condition: LCID.EQ.0: Constant velocity vector specified with VX, VY, and VZ. LCID.NE.0: VX, VY, and VZ give the velocity vector that is scaled by LCID.

Segment Set Card with Rotation. Card 1 format used when the SEGMENT_SET keyword option is active *with* the ROTATE keyword option.

Card 1d	1	2	3	4	5	6	7	8
Variable	SSID	LCID	XP	YP	ZP	NX	NY	NZ
Type	I	I	F	F	F	F	F	F
Default	none	0	0.0	0.0	0.0	0.0	0.0	0.0

VARIABLE	DESCRIPTION
SSID	ID of the segment set created with *DUALCESE_SEGMENTSET
LCID	Load curve ID for specifying the rotating speed frequency in Hz. This input is required.
XP, YP, ZP	Coordinates for a point on the axis of rotation
NX, NY, NZ	Unit vector for specifying the direction of the axis of rotation. This is not used for the 2D case.

Remarks:

1. **Boundary Movement Restrictions.** In this solid-wall condition, the boundary movement can only be in the tangential direction of the wall and should not affect the fluid domain size and mesh during the calculation. Otherwise an FSI

or moving mesh solver should be used. Also, this moving boundary condition only affects viscous flows (no-slip boundary condition).

2. **Fixed Solid Wall Boundary Condition.** If LCID = 0 and Vx = Vy = Vz = 0.0 (default), this will be a regular solid wall boundary condition.

*DUALCESE_CONTROL_LIMITER

Purpose: Sets some stability parameters used in the dual CESE compressible flow solver on the current dual CESE model.

Card 1	1	2	3	4	5	6	7	8
Variable	IDLMT	ALPHA	BETA	EPSR				
Type	I	F	F	F				
Default	0	0.0	0.0	0.0				
Remarks		1	2	3				

VARIABLE	DESCRIPTION
IDLMT	Set the stability limiter option (see dual CESE theory manual): EQ.0: Limiter format 1 (re-weighting) EQ.1: Limiter format 2 (relaxing)
ALPHA	Re-weighting coefficient, α (see dual CESE theory manual). Must be ≥ 0.
BETA	Numerical viscosity control coefficient, β (see dual CESE theory manual). $0 \leq \beta \leq 1$.
EPSR	Stability control coefficient, ε (see dual CESE theory manual). Must be ≥ 0.

Remarks:

1. **Re-weighting Coefficient.** Larger values of α give more stability, but less accuracy. Usually $\alpha = 2.0$ or 4.0 will be enough for normal shock problems.

2. **Numerical Viscosity Control Coefficient.** Larger values of β give more stability. For problems with shock waves, $\beta = 1.0$ is recommended.

3. **Stability Control Coefficient.** Larger values of ε give more stability, but less accuracy.

*DUALCESE_CONTROL_MESH_MOV

Purpose: Specify the algorithm for calculating the mesh movement (morphing) of a given DUALCESE part in an FSI problem. This keyword is for the moving mesh version of dual CESE.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	IALG	NINTER	RELERR	MXDISPR			
Type	I	I	I	F	F			
Default	none	9	100	10^{-3}	10^{-2}			

VARIABLE	DESCRIPTION
ID	ID for this mesh motion algorithm
IALG	Mesh motion algorithm: EQ.9: IDW scheme (default)
NITER	Number of linear solver iterations (when using a linear solver specified in IALG). No linear solvers have been implemented at this time, so this field is ignored.
RELERR	Relative error for determining convergence when using a linear solver specified in IALG. No linear solvers have been implemented at this time, so this field is ignored.
MXDISPR	Maximum displacement relative to element size to use as a criterion for avoiding the full calculation of the motion of the DUALCESE part on a given time step. If the full calculation can be avoided, the elements touching an FSI interface are still morphed, but it is assumed that this approximation will not lead to elements that are overly distorted.

*DUALCESE_CONTROL_SOLVER

Purpose: Set general purpose control variables for the dual CESE compressible flow solver.

> **WARNING:** As of version R14, this keyword is deprecated. Instead, in order to select the appropriate equations to solve, you should use *DUALCESE_SOLVER_SELECTION along with the corresponding *DUALCESE_-SOLVER_... card. Also, instead of this card, use *DU-ALCESE_MESH_GEOM for the geometry-related details.

Card 1	1	2	3	4	5	6	7	8
Variable	EQNS	IGEOM	IFRAME	MIXTYPE	IDC	ISNAN		
Type	A	A	A	A	F	I		
Default	EULER	none	FIXED	optional	0.25	0		
Remarks		1			2			

VARIABLE	DESCRIPTION
EQNS	Select the equations being solved with the dual CESE solver: EQ.NS: Navier-Stokes equations EQ.EULER: Euler equations
IGEOM	Sets the geometric dimension: EQ.2D: Two-dimensional (2D) problem EQ.3D: Three-dimensional (3D) problem EQ.AXI: 2D axisymmetric
IFRAME	Choose the frame of reference: EQ.FIXED: Usual non-moving reference frame (default). EQ.ROT: Non-inertial rotating reference frame. IFRAME = ROTATING may also be used.

VARIABLE	DESCRIPTION
MIXTYPE	Select the mix or multiphase model solver (if any): EQ.<blank>: No mix or multiphase model (default) EQ.HYBRID: Hybrid multiphase model solver EQ.TWO-PHASE: Two-phase multiphase solver
IDC	Contact interaction detection coefficient (for FSI and conjugate heat transfer problems)
ISNAN	Flag to check for NaN in the dual CESE solver solution arrays at the completion of each time step. This option can be useful for debugging purposes. There is a cost overhead when this option is active. EQ.0: No checking. EQ.1: Checking is active.

Remarks:

1. **Mesh and Boundary Conditions for 2D Problems.** If you want to use the 2D (IGEOM = 2D) or 2D axisymmetric (IGEOM=AXI) solver, the mesh should only be distributed in the *xy*-plane with the boundary conditions given only at the *xy* domain boundaries. Otherwise, a warning message will be given, and the 3D solver will be triggered instead.

   The 2D axisymmetric case will work only if the 2D mesh and corresponding boundary conditions are properly defined with the *x* and *y* coordinates corresponding to the radial and axial directions, respectively.

2. **Contact Interaction Detection Coefficient.** IDC is the same type of variable that is input on the *ICFD_CONTROL_FSI card. For an explanation, see Remark 1 for the *ICFD_CONTROL_FSI card.

3. **Dual CESE Solver and Restarts.** The dual CESE solver is *not* currently supported for restarts.

*DUALCESE_CONTROL_TIMESTEP

Purpose: Sets the time step control parameters for the CESE compressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	IDDT	CFL	DTINT					
Type	I	F	F					
Default	0	0.9	10^{-3}					

VARIABLE	DESCRIPTION
IDDT	Sets the time step option: EQ.0: Fixed time step size of DTINT, the given initial time step size NE.0: The time step size will be calculated based on the given CFL-number and the flow solution at the previous time step.
CFL	CFL number (Courant–Friedrichs–Lewy condition). $0.0 < \text{CFL} \leq 1.0$
DTINT	Initial time step size

*DUALCESE_DATABASE_HISTORY_ELEMENT_SET

Purpose: Enable output of dual CESE solver data at the element centroids of the selected dual CESE mesh elements. This element output goes to binary database binout.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	ESID	DT	LCUR	IOOPT				
Type	I	F	I	I				
Default	0	0.	0	0				

VARIABLE	DESCRIPTION
ESID	ID of a dual CESE element set (see *DUALCESE_ELEMENTSET)
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve. The abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_n)$. This is the default behavior. EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1})$. EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.

*DUALCESE_DATABASE_HISTORY_GLOBALS

Purpose: Enable output of global dual CESE solver data. The output goes to binary database binout.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	DT	LCUR	IOOPT					
Type	F	I	I					
Default	0.0	0	0					

VARIABLE	DESCRIPTION
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_n)$. This is the default behavior. EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1})$. EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.

*DUALCESE_DATABASE_HISTORY_NODE_SET

Purpose: Enable output of dual CESE solver data at the selected dual CESE mesh nodes. This nodal output goes to binary database binout.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	NSID	DT	LCUR	IOOPT				
Type	I	F	I	I				
Default	0	0.	0	0.				

VARIABLE	DESCRIPTION
NSID	ID of a dual CESE node set (see *DUALCESE_NODESET)
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_n)$. This is the default behavior. EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1})$. EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.

*DUALCESE_DATABASE_HISTORY_POINT_SET

Purpose: Enable output of dual CESE solver data at the selected points inside the dual CESE mesh. This point-based output goes to binary database binout.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	PNTSID	DT	LCUR	IOOPT				
Type	I	F	I	I				
Default	0	0.	0	0.				

VARIABLE	DESCRIPTION
PNTSID	ID of a dual CESE point set (see *DUALCESE_POINTSET)
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR: EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_n)$. This is the default behavior. EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1})$. EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.

*DUALCESE_DATABASE_HISTORY_SEGMENT_SET

Purpose: Enable output of dual CESE solver data averaged on each of the specified dual CESE segment sets. This segment set-based output goes to binary database binout. Multiple instances of this keyword may appear in the input.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	DT	LCUR	IOOPT					
Type	F	I	I					
Default	0.0	0	0					

Segment Sets Card. Define as many cards as necessary. Input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	SSID1	SSID2	SSID3	SSID4	SSID5	SSID6	SSID7	SSID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
DT	Time interval between outputs. If DT is zero, no output is generated.
LCUR	Optional curve ID specifying the time interval between outputs. Use *DEFINE_CURVE to define the curve; the abscissa is time, and the ordinate is time interval between outputs.
IOOPT	Flag to govern behavior of the output frequency load curve defined by LCUR:

VARIABLE	DESCRIPTION
	EQ.1: When output is generated at time t_n, the next output time t_{n+1} is computed as $$t_{n+1} = t_n + \mathrm{LCUR}(t_n) .$$ This is the default behavior.
	EQ.2: When output is generated at time t_n, the next output time t_{n+1} is computed as $$t_{n+1} = t_n + \mathrm{LCUR}(t_{n+1}) .$$
	EQ.3: Output is generated for each abscissa point in the load curve definition. The actual value of the load curve is ignored.
SSID*i*	i^{th} dual CESE segment set ID (see *DUALCESE_SEGMENTSET). For each of these segment sets, an average value of each dual CESE output variable is output to the binout file at the times selected by the fields in Card 1.

*DUALCESE_D3PLOT

Purpose: Specify the flow variables to be added to the dual CESE d3plot output.

Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	FLOW_VAR							
Type	A							

VARIABLE	DESCRIPTION
FLOW_VAR	Name of a flow variable to output to the d3plot file. The currently supported variables are listed in the table below.

Flow Variables:

This table lists the supported flow variables.

FLOW_VAR	DESCRIPTION
DENSITY	Density
VELOCITY	Velocity
MOMENTUM	Momentum
VORTICITY	Vorticity
TOTAL_ENERGY	Total energy
INTERNAL_ENERGY	Internal energy
PRESSURE	Pressure
TEMPERATURE	Temperature
MACH_NUMBER	Flow Mach number
SCHLIEREN_NUMBER	Quantity for capturing or highlighting the shock structure in a compressible flow

FLOW_VAR	DESCRIPTION
VOID_FRACTION	Void fraction (only for the dual CESE cavitation solver)
VOLUME_FRACTION	Volume fraction of the different materials in a hybrid or two-phase multiphase model
REACTANT_MASS_FRACTION	Mass fraction of the reactant (material α) with respect to the explosive material (material 2) in a hybrid multiphase model
LIQUID_MASS_FRACTION	Mass fraction of the liquid component of the fluid in the phase change model
VAPOR_MASS_FRACTION	Mass fraction of the vapor component of the fluid in the phase change model
OTHER_GASES_MASS_FRACTION	Mass fraction of the other gases in the fluid in the phase change model

*DUALCESE_D3PLOT_FLUID_SSID

Purpose: Generate surface d3plot output for the dual CESE solver on a specified dual CESE mesh segment set. These surfaces may be on the outside of the dual CESE fluid mesh that is in contact with the structural volume element parts.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							

VARIABLE	DESCRIPTION
SSID	ID of a segment set created with *DUALCESE_SEGMENTSET

Dual CESE variables to output. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	FLOW_VAR							
Type	A							

VARIABLE	DESCRIPTION
FLOW_VAR	Name of a flow variable to output to the d3plot file. The currently supported variables are listed in the table below.

Flow Variables:

This table lists the supported flow variables.

FLOW_VAR	DESCRIPTION
DENSITY	Density
VELOCITY	Velocity
MOMENTUM	Momentum

FLOW_VAR	DESCRIPTION
VORTICITY	Vorticity
TOTAL_ENERGY	Total energy
INTERNAL_ENERGY	Internal energy
PRESSURE	Pressure
TEMPERATURE	Temperature
MACH_NUMBER	Flow Mach number
SCHLIEREN_NUMBER	Quantity for capturing or highlighting the shock structure in a compressible flow
VOID_FRACTION	Void fraction (only for the dual CESE cavitation solver)
VOLUME_FRACTION	Volume fraction of the different materials in a hybrid or two-phase multiphase model
REACTANT_MASS_FRACTION	Mass fraction of the reactant (material α) with respect to the explosive material (material 2) in a hybrid multiphase model

*DUALCESE_ELE2D

Purpose: Define three and four node elements.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4				
Type	I	I	I	I	I	I				
Default	none	none	none	none	none	none				

VARIABLE	DESCRIPTION
EID	Element ID. Choose a unique number with respect to other elements.
PID	Part ID, see *DUALCESE_MESH_PART.
N1	Nodal point 1
N2	Nodal point 2
N3	Nodal point 3
N4	Nodal point 4

*DUALCESE_ELE3D

Purpose: Define three-dimensional fluid volume elements. These can be 4 node tetrahedra, 5 node pyramids, 6 node wedges (prisms), and 8 node hexahedra.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none	none	none
Remarks	1									

VARIABLE	DESCRIPTION
EID	Element ID. A unique number must be chosen.
PID	Part ID, see *DUALCESE_MESH_PART.
N1	Nodal point 1
N2	Nodal point 2
N3	Nodal point 3
⋮	⋮
N8	Nodal point 8

Remarks:

1. **Node Numbering.** Four, five, six, and eight node elements are allowed as numbered below. This ordering must be followed, or code termination will occur during the initialization phase with a negative volume message. In the case of a pyramid element, the base of the pyramid must follow the ordering used for the hexahedron. See *ELEMENT_SOLID for a figure showing the positions of the nodes in 4, 6, and 8 node elements.

4-noded tetrahedron	N1, N2, N3, N4, N4, N4, N4, N4
5-noded pyramid	N1, N2, N3, N4, N5, N5, N5, N5

6-noded pentahedron	N1, N2, N3, N4, N5, N5, N6, N6
8-noded hexahedron	N1, N2, N3, N4, N5, N6, N7, N8

*DUALCESE_ELEMENTSET

Purpose: Define a set of dual CESE mesh elements.

Card 1	1	2	3	4	5	6	7	8
Variable	ESID							
Type	I							
Default	none							

Element ID Cards. List of elements in the set, where the element IDs are defined with *DUALCESE_ELE2D or *DUALCESE_ELE3D cards. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	EID1	EID2	EID3	EID4	EID5	EID6	EID7	EID8
Type	I	I	I	I	I	I	I	I

VARIABLE	DESCRIPTION
ESID	Set ID. All dual CESE element sets should have a unique set ID.
EID*i*	Element ID *i*

*DUALCESE_EOS_CAV_HOMOG_EQUILIB

Purpose: Define the coefficients in the equation of state (EOS) for the homogeneous equilibrium cavitation model.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	RHOVAP	RHOLIQ	AVAP	ALIQ	MUVAP	MULIQ	PSATVAP
Type	I	F	F	F	F	F	F	F
Default	none	0.8	880.0	334.0	1386.0	1.435e-5	1.586e-4	1.2e+4

VARIABLE	DESCRIPTION
EOSID	Equation of state ID
RHOVAP	Density of the saturated vapor, ρ_{vap}
RHOLIQ	Density of the saturated liquid, ρ_{liq}
AVAP	Sound speed of the saturated vapor, a_{vap}
ALIQ	Sound speed of the saturated liquid, a_{liq}
MUVAP	Dynamic viscosity of the vapor, μ_{vap}
MULIQ	Dynamic viscosity of the liquid, μ_{liq}
PSATVAP	Pressure of the saturated vapor, P_{SatVap}

Remarks:

1. **EOS Validity.** In this homogeneous equilibrium cavitation model, a barotropic equation of state is used. This model can be used in small scale and high-speed cavitation flows, but it is not good for large-scale, low-speed cavitation calculations.

*DUALCESE_EOS_COCHRAN_CHAN

Purpose: Define a Cochran-Chan type of EOS that provides a means to represent a condensed phase explosive in a dual CESE multiphase model.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	A	B	EPS1	EPS2	GAMMA0	RHO0	E0
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	0.0

Card 2	1	2	3	4	5	6	7	8
Variable	CV							
Type	F							
Default	none							

VARIABLE	DESCRIPTION
EOSID	Equation of state ID for the dual CESE solver
A	Model parameter (in pressure units), A
B	Model parameter (in pressure units), B
EPS1	Model constant (dimensionless), ε_1
EPS2	Model constant (dimensionless), ε_2
GAMMA0	Gruneisen coefficient
RHO0	Initial or reference density, ρ_0
E0	Represents the heat of detonation released during the reactions, or the constant rate of afterburn energy added (E0 = 0.0 is the default), e_0
CV	Heat capacity, C_v

Remarks:

The Cochran-Chan EOS like the JWL EOS (*DUALCESE_EOS_JWL) is a type of Mie-Gruneisen EOS. The equations of state of a Mie-Gruneisen form are given by:

$$P(\rho, e) = P_{\text{ref}} + \Gamma(\rho)\rho[e - e_{\text{ref}}(\rho)]$$

Here $\Gamma(\rho)$ is the Gruneisen coefficient. For the Cochran-Chan EOS reference pressure and energy are given by:

$$P_{ref}(\rho) = A\left(\frac{\rho_0}{\rho}\right)^{-\varepsilon_1} - B\left(\frac{\rho_0}{\rho}\right)^{-\varepsilon_2}$$

$$e_{\text{ref}}(\rho) = \frac{A}{\rho_0(1-\varepsilon_1)}\left(\frac{\rho_0}{\rho}\right)^{1-\varepsilon_1} + \frac{B}{\rho_0(1-\varepsilon_2)}\left(\frac{\rho_0}{\rho}\right)^{1-\varepsilon_2} - e_0$$

*DUALCESE_EOS_COOLPROP

Purpose: Define an equation of state (EOS) to be evaluated using the COOLPROP EOS library

Note that the COOLPROP library is not provided by ANSYS. You need to download a 64-bit version of the shared library from a public repository, such as:

https://sourceforge.net/projects/coolprop/files/CoolProp/6.3.0/shared_library/Linux/64bit/

> **WARNING:** Since the *MODULE capability is not yet working in the Windows build of LS-DYNA, do *not* attempt to use a Windows DLL version of the COOLPROP shared library.

To use the COOLPROP shared library with this keyword card, load this shared library into LS-DYNA using the *MODULE capability. The following *MODULE card needs to appear before a *DUALCESE_MODEL card (not inside the file hierarchy of any file specified with a *DUALCESE_MODEL card):

```
*MODULE_LOAD
UserA                    DUALCESE COOLPROP
 < path to installed COOLPROP shared library >
```

Card Summary:

Card 1. This card is required.

EOSID	NCOMP	TYPE	PHASE	TABULAR			

Card 2. Include as many cards as needed to specify mole fractions for the NCOMP components of the fluid.

MOL_FR1	MOL_FR2	MOL_FR3	MOL_FR4	MOL_FR5	MOL_FR6	MOL_FR7	MOL_FR8

Card 3. Include this card when the TABULAR field is active on Card 1.

N_T	N_DEN	DEN_LOW	DEN_HIGH	T_LOW	T_HIGH		

Card 4. This card is required.

FLUIDNAME

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	NCOMP	TYPE	PHASE	TABULAR			
Type	I	I	A	A	A			
Default	none	none	none	GAS	optional			

VARIABLE	DESCRIPTION
EOSID	ID for this EOS
NCOMP	Number of components in the fluid composition
TYPE	Fluid type: EQ.PURE: A single component fluid (default) EQ.PSEUDOPURE: A predefined fluid mixture EQ.MIXTURE: A fluid mixture with NCOMP components
PHASE	Phase of the fluid. EQ.GAS: Gas phase EQ.LIQUID: Liquid phase
TABULAR	Type of lookup tables to build for this EOS: EQ.<BLANK>: No table lookup (default) EQ.P_EIN: Build tables of pressure and internal energy, both as a function of density and temperature.

COOLPROP Parameters by Fluid Component. Repeat this card as many times as needed to input mole fractions for the NCOMP components of the fluid.

Card 2	1	2	3	4	5	6	7	8
Variable	MOL_FR1	MOL_FR2	MOL_FR3	MOL_FR4	MOL_FR5	MOL_FR6	MOL_FR7	MOL_FR8
Type	F	F	F	F	F	F	F	F

VARIABLE	DESCRIPTION
MOL_FR*i*	Mole fraction of the i^{th} component

COOLPROP EOS Table Density and Temperature Ranges. This card is included when the TABULAR option on Card 1 is active.

Card 3	1	2	3	4	5	6	7	8
Variable	N_T	N_DEN	DEN_LOW	DEN_HIGH	T_LOW	T_HIGH		
Type	I	I	F	F	F	F		
Default	none	none	none	none	none	none		
Remarks	1	1	2	2	2	2		

VARIABLE	DESCRIPTION
N_T	Number of temperature values in the tables
N_DEN	Number of density values (on a log scale) in the tables
DEN_LOW	Minimum density available in the tables (in model units)
DEN_HIGH	Maximum density available in the tables (in model units)
T_LOW	Minimum temperature available in the tables (in model units)
T_HIGH	Maximum temperature available in the tables (in model units)

Name of CoolProp fluid. This card is required.

Card 4	1	2	3	4	5	6	7	8
Variable	FLUIDNAME							
Type	A							

VARIABLE	DESCRIPTION
FLUIDNAME	Name of a fluid that has an EOS in CoolProp. For a list of the supported pure and pseudo-pure fluids, see: http://www.coolprop.org/fluid_properties/PurePseudoPure.html#list-of-fluids Note that the predefined fluid mixtures are not supported at this time.

Remarks:

1. **Number of Values in the Lookup Tables.** The number of density and temperature values in the tables should not be too few to give good resolution of the EOS. Note that the cost of building the EOS from these tables rises with these numbers, as well as the computer memory required. Nevertheless, if these numbers are too small (< 20), then the accuracy may suffer, while larger numbers of density and temperature points improves the accuracy.

2. **Valid Value Ranges for the Lookup Tables.** For many equations of state in the CoolProp library, there is a range of valid densities and temperatures. Thus, the low and high limits for the table densities and temperatures should not lie outside these ranges. Please refer to the CoolProp documentation for that information.

*DUALCESE_EOS_IDEAL_GAS

Purpose: Define the coefficients $C_v$and C_p in the equation of state for an ideal gas in the dual CESE fluid solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	CV	CP	E0				
Type	I	F	F	F				
Default	none	717.5	1004.5	0.0				
Remarks		1	1	2				

VARIABLE	DESCRIPTION
EOSID	Equation of state ID
CV	Specific heat at constant volume, C_v
CP	Specific heat at constant pressure, C_p
E0	Represents the heat of detonation released during the reactions, or the constant rate of afterburn energy added (E0 = 0.0 is the default), e_0

Remarks:

1. **Units.** As with other solvers in LS-DYNA, you are responsible for unit consistency. For example, if you want to use dimensionless variables, CV and CP should also be replaced by the corresponding dimensionless ones. If the dual CESE model has a specified system of units either directly from the *DUALCESE_MODEL card or inherited from the overall problem input, then these values need to be given in that unit system.

2. **E0.** E0 is used only with the hybrid multiphase solver where the EOS of the reactant is specified by this ideal gas EOS in the *DUALCESE_EOS_SET card.

*DUALCESE_EOS_INFLATOR1

Purpose: Define an EOS using C_p and C_v thermodynamic expansions for an inflator gas mixture with a single temperature range.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID							
Type	I							
Default	none							

Card 2	1	2	3	4	5	6	7	8
Variable	CP0	CP1	CP2	CP3	CP4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Card 3	1	2	3	4	5	6	7	8
Variable	CV0	CV1	CV2	CV3	CV4			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

VARIABLE	DESCRIPTION
EOSID	Equation of state ID for the dual CESE solver
CP0, …, CP4	Coefficients of temperature-dependent specific heat at constant pressure $$C_p(T) = C_{p_0} + C_{p_1}T + C_{p_2}T^2 + C_{p_3}T^3 + C_{p_4}T^4$$

VARIABLE	DESCRIPTION
CV0, …, CV4	Coefficients of temperature-dependent specific heat at constant volume $$C_v(T) = C_{v0} + C_{v1}T + C_{v2}T^2 + C_{v3}T^3 + C_{v4}T^4$$

Remarks:

These coefficient expansions for the specific heats over the entire temperature range are generated by the zero-dimensional inflator model solver. See *CHEMISTRY_CONTROL_INFLATOR and *CHEMISTRY_INFLATOR_PROPERTIES for details related to running that solver.

*DUALCESE_EOS_INFLATOR2

Purpose: Define an EOS using C_p and C_v thermodynamic expansions for an inflator gas mixture with two temperature ranges, one below 1000 Kelvin, and the other above 1000 Kelvin.

Card Summary:

Card 1. This card is required.

EOSID							

Card 2. This card is required. This card with Card 3 specifies C_p. This card gives the coefficients for T < 1000 K.

CP10	CP11	CP12	CP13	CP14			

Card 3. This card is required. This card gives the coefficients for T > 1000 K.

CP20	CP21	CP22	CP23	CP24			

Card 4. This card is required. This card with Card 5 defines C_v. This card gives the coefficients for T < 1000 K.

CV10	CV11	CV12	CV13	CV14			

Card 5. This card is required. This card gives the coefficients for T > 1000 K.

CV20	CV21	CV22	CV23	CV24			

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
EOSID	Equation of state ID for the dual CESE solver

Coefficients for the expansion to determine specific heat at constant pressure for T < 1000 K.

Card 2	1	2	3	4	5	6	7	8
Variable	CP10	CP11	CP12	CP13	CP14			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Coefficients for the expansion to determine specific heat at constant pressure for T > 1000 K.

Card 3	1	2	3	4	5	6	7	8
Variable	CP20	CP21	CP22	CP23	CP24			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

VARIABLE	DESCRIPTION
CP10, …, CP14	Coefficients of temperature-dependent specific heat at constant pressure valid for T < 1000 K
CP20, …, CP24	Coefficients of temperature-dependent specific heat at constant pressure valid for T > 1000 K

Cards 2 and 3 give C_p over the two temperature ranges:

$$C_p(T) = \begin{cases} \text{CP10} + \text{CP11} \times T + \text{CP12} \times T^2 + \text{CP13} \times T^3 + \text{CP14} \times T^4 & \text{for } T < 1000 \text{ K} \\ \text{CP20} + \text{CP21} \times T + \text{CP22} \times T^2 + \text{CP23} \times T^3 + \text{CP24} \times T^4 & \text{for } T > 1000 \text{ K} \end{cases}$$

Card for the Expansion of Specific Heat at Constant Volume. Valid for T < 1000 K

Card 4	1	2	3	4	5	6	7	8
Variable	CV10	CV11	CV12	CV13	CV14			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

Card for the Expansion of Specific Heat at Constant Volume. Valid for T > 1000 K

Card 5	1	2	3	4	5	6	7	8
Variable	CV20	CV21	CV22	CV23	CV24			
Type	F	F	F	F	F			
Default	0.	0.	0.	0.	0.			

VARIABLE	DESCRIPTION
CV10, …, CV14	Coefficients of temperature-dependent specific heat at constant volume valid for T < 1000 K
CV20, …, CV24	Coefficients of temperature-dependent specific heat at constant volume valid for T > 1000 K

Cards 4 and 5 give C_v over the two temperature ranges:

$$C_v(T) = \begin{cases} \text{CV10} + \text{CV11} \times T + \text{CV12} \times T^2 + \text{CV13} \times T^3 + \text{CV14} \times T^4 & \text{for } T < 1000 \text{ K} \\ \text{CV20} + \text{CV21} \times T + \text{CV22} \times T^2 + \text{CV23} \times T^3 + \text{CV24} \times T^4 & \text{for } T > 1000 \text{ K} \end{cases}$$

Remarks:

These coefficient expansions for the specific heats over two temperature ranges are generated by the zero-dimensional inflator model solver. See *CHEMISTRY_CONTROL_INFLATOR and *CHEMISTRY_INFLATOR_PROPERTIES for details related to running that solver.

*DUALCESE_EOS_JWL

Purpose: Define a JWL-type EOS that provides a means to represent a condensed phase explosive in a dual CESE multiphase model.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	A	B	R1	R2	GAMMA0	RHO0	E0
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	0.0

Card 2	1	2	3	4	5	6	7	8
Variable	CV							
Type	F							
Default	none							

VARIABLE	DESCRIPTION
EOSID	Equation of state ID for the dual CESE solver.
A	Model parameter, A (in pressure units)
B	Model parameter, B (in pressure units)
R1	Model constant, R_1 (dimensionless)
R2	Model constant, R_2 (dimensionless)
GAMMA0	Gruneisen coefficient
RHO0	Initial or reference density, ρ_0
E0	Represents the heat of detonation released during the reactions or the constant rate of afterburn energy added (E0 = 0.0 for standard JWL EOS), e_0
CV	Heat capacity, C_v

Remark:

The equations of state of a Mie-Gruneisen form are given by:

$$P(\rho, e) = P_{\text{ref}} + \Gamma(\rho)\rho[e - e_{\text{ref}}(\rho)]$$

Here $\Gamma(\rho)$ is the Gruneisen coefficient. Equations of state of this type are very popular in condensed phase explosive modeling. Depending on the form of the reference pressure and energy functions, different EOS types can be retrieved. The JWL EOS is one type with reference pressure and energy given by:

$$P_{\text{ref}}(\rho) = A\exp\left(\frac{-R_1\rho_0}{\rho}\right) + B\exp\left(\frac{-R_2\rho_0}{\rho}\right)$$

$$e_{\text{ref}}(\rho) = \frac{A}{\rho_0 R_1}\exp\left(\frac{-R_1\rho_0}{\rho}\right) + \frac{B}{\rho_0 R_2}\exp\left(\frac{-R_2\rho_0}{\rho}\right) - e_0$$

*DUALCESE_EOS_NASG

Purpose: Define a Noble-Abel Stiffened-Gas (NASG) type fluid EOS for use in the phase-change dual CESE solver. See M'etayer and Saurel [2016] for details.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	GAMMA	CV	PINF	Q	QP	B	W
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
EOSID	Equation of state ID
GAMMA	Ratio of specific heats, γ
CV	Specific heat at constant volume
PINF	Parameter for a given phase, p_∞ (dimensions: $\left[\frac{M}{LT^2}\right]$). See Le M'etayer and Saurel [2016].
Q	Parameter for a given phase, q (dimensions: $\left[\frac{L^2}{T^2}\right]$). q is the heat bond for the phase. See Le M'etayer and Saurel [2016].
QP	Parameter for a given phase, q' (dimensions: $\left[\frac{L^2}{T^2\theta}\right]$, where θ represents the dimension of temperature). See Le M'etayer and Saurel [2016].
B	Parameter for a given phase, b (dimensions: $\left[\frac{L^3}{M}\right]$). b indicates the fluid's covolume. See Le M'etayer and Saurel [2016].
W	Molar mass of this fluid

Remarks:

The NASG EOS given by M'etayer and Saurel [2016] has the following form:

$$p(\nu, e) = \frac{(\gamma - 1)(e - q)}{\nu - b} - \gamma p_\infty$$
$$T(p, \nu) = \frac{(\nu - b)(p + p_\infty)}{C_v(\gamma - 1)}$$
$$g(p, T) = (\gamma C_v - q')T - C_v T \ln\left[\frac{T^\gamma}{(p + p_\infty)^{\gamma-1}}\right] + bp + q$$
$$c(p, \nu) = \sqrt{\gamma\nu(p + p_\infty)}$$

where γ, C_v, p_∞, q, q' and b are parameters needed for each phase. ν, e, g and c are for each fluid phase the specific volume, the specific internal energy, the specific Gibbs free energy, and the sound speed, respectively. $g = h - Ts$ with h and s as the specific enthalpy and specific entropy, respectively.

References:

[1] Le M'etayer, O. and R. Saurel, "The Nobel-Abel Stiffened-Gas equation of state," *Physics of Fluids*, 28 (4), (2016).

*DUALCESE_EOS_REFPROP

Purpose: Define an equation of state (EOS) to be evaluated using the REFPROP EOS library

Note that the REFPROP library is provided by ANSYS; it is REFPROP v10.0 from NIST.

> **WARNING:** Since the *MODULE capability is not yet working in the Windows build of LS-DYNA, do *not* attempt to use a Windows DLL version of the REFPROP shared library that comes with REFPROP v10.0.

To use the REFPROP shared library with this keyword card, load this shared library into LS-DYNA using the *MODULE capability. The following *MODULE card needs to appear before a *DUALCESE_MODEL card (not inside the file hierarchy of any file specified with a *DUALCESE_MODEL card):

```
*MODULE_LOAD
UserA                    DUALCESE REFPROP
 < path to the installed REFPROP shared library >
```

In addition, for REFPROP to be able to find the appropriate EOS data, *DUALCESE_EOS_REFPROP_PATH must also be given somewhere inside a *DUALCESE_MODEL file hierarchy to point to the place in your filesystem where REFPROP has been installed.

Card Summary:

Card 1. This card is required.

EOSID	NCOMP	TYPE	PHASE	TABULAR			

Card 2. Include as many cards as needed to specify mole fractions for the NCOMP components of the fluid.

MOL_FR1	MOL_FR2	MOL_FR3	MOL_FR4	MOL_FR5	MOL_FR6	MOL_FR7	MOL_FR8

Card 3. This card is included when the TABULAR field is active on Card 1.

N_T	N_DEN	DEN_LOW	DEN_HIGH	T_LOW	T_HIGH		

Card 4. This card is required.

FLUIDNAME

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	NCOMP	TYPE	PHASE	TABULAR			
Type	I	I	A	A	A			
Default	none	none	none	GAS	optional			

VARIABLE	DESCRIPTION
EOSID	ID for this EOS
NCOMP	Number of components in the fluid composition
TYPE	Fluid type: EQ.PURE: A single component fluid (default) EQ.PSEUDOPURE: A predefined fluid mixture EQ.MIXTURE: A fluid mixture with NCOMP components
PHASE	Phase of the fluid: EQ.GAS: Gas phase EQ.LIQUID: Liquid phase
TABULAR	Type of lookup tables to build for this EOS: EQ.<BLANK>: No table lookup (default) EQ.P_EIN: Build tables of pressure and internal energy, both as a function of density and temperature.

REFPROP Parameters by Fluid Component. Repeat this card as many times as needed to input mole fractions for the NCOMP components of the fluid.

Card 2	1	2	3	4	5	6	7	8
Variable	MOL_FR1	MOL_FR2	MOL_FR3	MOL_FR4	MOL_FR5	MOL_FR6	MOL_FR7	MOL_FR8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
MOL_FR*i*	Mole fraction of the i^{th} component of the fluid.

.

REFPROP EOS Table Density and Temperature Ranges. This card is included when the TABULAR field is active on Card 1.

Card 3	1	2	3	4	5	6	7	8
Variable	N_T	N_DEN	DEN_LOW	DEN_HIGH	T_LOW	T_HIGH		
Type	I	I	F	F	F	F		
Default	none	none	none	none	none	none		
Remarks	1	1	2	2	2	2		

VARIABLE	DESCRIPTION
N_T	Number of temperature values in the tables
N_DEN	Number of density values (on a log scale) in the table
DEN_LOW	Minimum density available in the tables (in model units)
DEN_HIGH	Maximum density available in the tables (in model units)
T_LOW	Minimum temperature available in the tables (in model units)
T_HIGH	Maximum temperature available in the tables (in model units)

Name of REFPROP fluid. This card is required.

Card 4	1	2	3	4	5	6	7	8
Variable	FLUIDNAME							
Type	A							

VARIABLE	DESCRIPTION
FLUID-NAME	Name of a fluid that has an EOS in REFPROP. For a list of the supported pure and pseudo-pure fluids, see the directory of supported fluids that comes with the REFPROP v10.0 library from ANSYS. Note that the predefined fluid mixtures are not supported at this time.

Remarks:

1. **Number of Values in the Lookup Table.** The number of values of density and temperature axes of the tables should not be too few to give good resolution of the EOS. Note that the cost of building the EOS from these tables rises with these numbers, as well as the computer memory required. Nevertheless, if these numbers are too small (< 20), then the accuracy may suffer, while larger numbers of density and temperature points improves the accuracy.

2. **Valid Value Ranges for the Lookup Tables.** For many equations of state in the REFPROP library, a range of densities and temperatures are valid. Thus, the low and high limits for the table densities and temperatures should not lie outside these ranges. Please refer to the REFPROP documentation for that information.

*DUALCESE_EOS_REFPROP_PATH

Purpose: Provide the file path to the directory where the REFPROP EOS system is installed.

Note that in any problem where a *DUALCESE_EOS_REFPROP card is used, you must also provide a *DUALCESE_EOS_REFPROP_PATH card somewhere inside a *DUALCESE_MODEL file hierarchy to point to the place in your filesystem where REFPROP has been installed so that the appropriate EOS data can be loaded.

Card 1	1	2	3	4	5	6	7	8
Variable	PATH							
Type	A							

VARIABLE	DESCRIPTION
PATH	Path giving the directory where the REFPROP data is installed.

*DUALCESE_EOS_SET

Purpose: Define a set of equations of state that are used together to compute the thermodynamic state of a multiphase fluid for the dual CESE solver.

> **WARNING:** As of version R14, this keyword is deprecated. Instead, use *DUALCESE_SOLVER_HYBRID_MULTIPHASE for the hybrid multiphase case and *DUALCESE_-SOLVER_TWO-PHASE_MULTIPHASE for the two-phase multiphase case.

Include one card for each dual CESE multiphase mesh. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSSID	EOSINID	EOSRCTID	EOSPRDID				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
EOSSID	Set ID of the EOS mixture of a given multiphase fluid
EOSINID	EOS ID of the inert component of the multiphase mixture
EOSRCTID	EOS ID of the reactant phase of the multiphase mixture
EOSPRDID	EOS ID of the product phase of the multiphase mixture

*DUALCESE_EOS_STIFFENED_GAS

Purpose: Define a stiffened gas type fluid EOS for use by the dual CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	GA	BT					
Type	I	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
EOSID	Equation of state ID for this dual CESE solver EOS
GA	Adiabatic exponent, γ. Must be > 1.0.
BT	Reference pressure, β. Must be ≥ 0.0

Remark:

The stiffened gas equation of state:

$$p(\rho, e) = (\gamma - 1)\rho e - \gamma\beta$$

provides a fundamental characterization of material properties of fluids. Here e is the internal energy per unit mass, ρ is the density, and γ and β are two thermodynamic constants. γ and β can be determined by a fitting procedure from laboratory data. A typical set of parameter values for water are: $\gamma = 7$ and $\beta = 3000$ atm while for human blood are: $\gamma = 5.527$ and $\beta = 614.6$ MPa. In addition to the modelling of a liquid, it is often used to describe other type of materials, including many compressible solids of practical importance.

*DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED

Purpose: Define a Van Der Waals generalized type fluid EOS for use in the dual CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	A	B	GA	BT			
Type	I	F	F	F	F			
Default	none	none	none	none	none			

VARIABLE	DESCRIPTION
EOSID	Equation of state ID
A	van der Waals gas constant for molecular cohesive forces, a
B	van der Waals gas constant for the finite size of molecules, b
GA	Ratio of specific heats, γ. Must be > 1.0.
BT	Reference pressure, β. Must be ≥ 0.0. See *DUALCESE_EOS_-STIFFENED_GAS.

Remark:

The generalized Van der Waals equation of state can be written as:

$$p(\rho, e) = \frac{(\gamma - 1)}{1 - b\rho}\left(\rho e - \beta + a\rho^2\right) - \left(\beta + a\rho^2\right)$$

Here e denotes the specific internal energy, γ is the ratio of specific heats ($\gamma > 1$), β is a reference pressure, and the quantities a and b are the van der Waals gas constants for molecular cohesive forces and the finite size of molecules, respectively ($a \geq 0$, $0 \leq b < 1/\rho$). This EOS is often used to deal with possible real-gas effect (without phase transition) when both the temperature and pressure are high. When $\beta = 0$, the generalized van der Waals equation of state becomes a standard Van der Waals equation of state. If $a = b = 0$, the stiffened gases EOS will be recovered.

References:

[1] G. Allaire, S. Clerc, S. Kokh, A five-equation model for the simulation of interfaces between compressible fluids. J. Comp. Phys. 181 (2) (2002) 577-616.

[2] K.M. Shyue, A fluid-mixture type algorithm for compressible multicomponent flow with van der Waals equation of state, J. Comp. Phys. 156, 43 (1999)

*DUALCESE_FSI_EXCLUDE

Purpose: Provide a list of mechanics solver parts that are not involved in the dual CESE FSI calculation. This keyword is intended for increasing computational efficiency by excluding parts that will not involve significant FSI interactions with the dual CESE compressible fluid solver.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
PID*n*	IDs of mechanics parts that will be excluded from the FSI interaction calculation with the dual CESE solver

*DUALCESE_INCLUDE_MODEL

Purpose: Provide the filename of a file containing additional keywords belonging to a dual CESE model. Any number of these *DUALCESE_INCLUDE_MODEL keywords may be used in a single dual CESE model, where at the top level the overall model begins with a *DUALCESE_MODEL card.

Card 1	1	2	3	4	5	6	7	8
Variable	FILENAME							
Type	A							

VARIABLE	DESCRIPTION
FILENAME	Filename of the keyword file containing more of the dual CESE model. This card is only allowed inside a file that is given in one instance of a *DUALCESE_MODEL keyword card.

*DUALCESE_INITIAL

Purpose: Specify constant initial conditions for flow variables at the centroid of each dual CESE fluid element.

Card 1	1	2	3	4	5	6	7	8
Variable	U	V	W	RH	P	T		IFUNC
Type	F	F	F	F	F	F		I
Default	0	0.0	0.0	1.225	0.0	0.0		none

VARIABLE	DESCRIPTION
U, V, W	x, y, and z velocity components, respectively
RHO	Density, ρ
P	Pressure, P
T	Temperature, T
IFUNC	Option to define initial conditions using *DEFINE_FUNCTION cards: EQ.0: Not in use. EQ.1: All values for initial velocity, pressure, density, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x,y,z), meaning that each variable's initial profile is a function of position.

Remarks:

1. **Required Input.** Usually, only two of ρ, P, and T are needed to be specified (besides the velocity). If all three are given, only ρ and P will be used.

2. **Applicable Elements.** These initial conditions will be applied in those elements that have not been assigned a value by *DUALCESE_INITIAL_*OPTION* cards for individual elements or sets of elements.

*DUALCESE_INITIAL_SET

Purpose: Specify initial conditions for the flow variables at the centroid of each element in an element subset of the dual CESE mesh.

Card 1	1	2	3	4	5	6	7	8
Variable	ESID	IFUNC						
Type	I	I						
Default	none	none						

Card 2	1	2	3	4	5	6	7	8
Variable	U	V	W	RHO	P	T		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
ESID	Element set ID (see *DUALCESE_ELEMENTSET)
IFUNC	Option to define initial conditions using *DEFINE_FUNCTION cards: EQ.0: Not in use. EQ.1: All values for initial velocity, pressure, density, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x,y,z), meaning that each variable's initial profile is a function of position.
U, V, W	x, y, and z velocity components, respectively
RHO	Density, ρ
P	Pressure, P

VARIABLE	DESCRIPTION
T	Temperature, T

Remarks:

1. **Required Input.** Usually, only two of ρ, P and T are needed to be specified (besides the velocity). If all three are given, only ρ and P will be used.

2. **Keyword Priority.** The priority of this card is higher than *DUALCESE_INITIAL. Thus, if an element is assigned an initial value by this card, *DUALCESE_-INITIAL will no longer apply to that element.

*DUALCESE_INITIAL_HYBRID

Purpose: Specifies values to use for velocity, pressure, etc. for problem initialization of a hybrid multiphase model on the dual CESE mesh. Note that these values can be overridden in some mesh elements by use of the *DUALCESE_INITIAL_HYBRID_SET card.

The hybrid multiphase model involves two materials, separated by a material surface. These materials do not mix. The first material is inert while the second material is an explosive mixture, composed of reactants and products. This model is useful for simulations with high explosives. See Michael and Nikiforakis 2016 for details about this model.

Card 1	1	2	3	4	5	6	7	8
Variable	Z1	RA	UIC	VIC	WIC	RHO1	RHO_A	RHO_B
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	PIC	TIC		IFUNC				
Type	F	F		I				
Default	none	none		none				

VARIABLE	DESCRIPTION
Z1	Volume fraction of material 1 (or color function). This is usually a value of 0 or 1. For numerical stability, however, use a very small value instead of zero.
RA	Mass fraction of the reactant (material α) with respect to material 2 (the explosive mixture)
UIC, VIC, WIC	Multiphase flow velocity components in the x, y, and z-directions, respectively
RHO1	Density of material 1

VARIABLE	DESCRIPTION
RHO_A	Density of the reactant (material α)
RHO_B	Density of the product (material β)
PIC	Equilibrium multifluid pressure
TIC	Equilibrium multifluid temperature
IFUNC	Option to define initial conditions using *DEFINE_FUNCTION cards: EQ.0: Not in use. EQ.1: All values for initial velocity, pressure, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x,y,z), meaning that each variable's initial profile is a function of position.

*DUALCESE_INITIAL_HYBRID_SET

Purpose: Specifies values to use for velocity, pressure, etc. for problem initialization of a hybrid multiphase model in an element subset of the dual CESE mesh. The values specified here override the values set on *DUALCESE_INITIAL_HYBRID for the element subset.

Card 1	1	2	3	4	5	6	7	8
Variable	ESID	IFUNC						
Type	I	I						
Default	none	none						

Card 2	1	2	3	4	5	6	7	8
Variable	Z1	RA	UIC	VIC	WIC	RHO1	RHO_A	RHO_B
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 3	1	2	3	4	5	6	7	8
Variable	PIC	TIC						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
ESID	Element set ID (see *DUALCESE_ELEMENTSET)
IFUNC	Option to define initial conditions using *DEFINE_FUNCTION cards:

VARIABLE	DESCRIPTION
	EQ.0: Not in use. EQ.1: All values for initial velocity, pressure, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x,y,z), meaning that each variable's initial profile is a function of position.
Z1	Volume fraction of material 1 (or color function). This is usually a value of 0 or 1. For numerical stability, however, use a small value instead of 0.
RA	Mass fraction of the reactant (material α) with respect to material 2 (the explosive mixture)
UIC, VIC, WIC	Multiphase flow velocity components in the x, y, and z-directions, respectively
RHO1	Density of material 1
RHO_A	Density of the reactant (material α)
RHO_B	Density of the product (material β)
PIC	Equilibrium multifluid pressure
TIC	Equilibrium multifluid temperature

*DUALCESE_INITIAL_PHASE_CHANGE

Purpose: Specify values to use for velocity, pressure, etc. for problem initialization of a phase-change multiphase model on the dual CESE mesh. Note that these values can be overridden in some mesh elements by use of the *DUALCESE_INITIAL_PHASE_CHANGE_SET card.

The phase-change multiphase model involves three fluids: a liquid, its vapor, and other gases.

Card 1	1	2	3	4	5	6	7	8
Variable	UIC	VIC	WIC	RHO	PIC	TIC	Y1	Y2
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	Y3			IFUNC				
Type	F			I				
Default	none			0				

VARIABLE	DESCRIPTION
UIC, VIC, WIC	Multiphase flow velocity components in the x, y, and z-directions, respectively
RHO	Mixture density
PIC	Equilibrium multifluid pressure
TIC	Equilibrium multifluid temperature
Y1	Mass fraction of fluid 1
Y2	Mass fraction of fluid 2

VARIABLE	DESCRIPTION
Y3	Mass fraction of fluid 3
IFUNC	Option to define initial conditions with *DEFINE_FUNCTION: EQ.0: Not in use. EQ.1: All values for initial velocity, pressure, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x, y, z). Thus, each variable's initial profile is a function of position.

*DUALCESE_INITIAL_PHASE_CHANGE_SET

Purpose: Specify values to use for velocity, pressure, etc. for problem initialization of a phase- change multiphase model in an element subset of the dual CESE mesh. The values specified here override the values set on *DUALCESE_INITIAL_PHASE_CHANGE for the element subset.

Card 1	1	2	3	4	5	6	7	8
Variable	ESID	IFUNC						
Type	I	I						
Default	none	0						

Card 2	1	2	3	4	5	6	7	8
Variable	UIC	VIC	WIC	RHO	PIC	TIC	Y1	Y2
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 3	1	2	3	4	5	6	7	8
Variable	Y3							
Type	F							
Default	none							

VARIABLE	DESCRIPTION
ESID	Element set ID (see *DUALCESE_ELEMENTSET)
IFUNC	Option to define initial conditions with *DEFINE_FUNCTION: EQ.0: Not in use.

VARIABLE	DESCRIPTION
	EQ.1: All values for initial velocity, pressure, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x, y, z). Thus, each variable's initial profile is a function of position.
UIC, VIC, WIC	Multiphase flow velocity components in the x, y, and z-directions, respectively
RHO	Mixture density
PIC	Equilibrium multifluid pressure
TIC	Equilibrium multifluid temperature
Y1	Mass fraction of fluid 1
Y2	Mass fraction of fluid 2
Y3	Mass fraction of fluid 3

*DUALCESE_INITIAL_TWO-PHASE

Purpose: Specify values to use for velocity, pressure, etc. for problem initialization of a two-phase multifluid model on the dual CESE mesh. Note that these values can be over-ridden in some mesh elements by use of the *DUALCESE_INITIAL_TWO-PHASE_SET card.

Card 1	1	2	3	4	5	6	7	8
Variable	Z1	UIC	VIC	WIC	RHO_1	RHO_2	PIC	TIC
Type	F	F	F	F	F	F	F	F
Default	none	none	0.	0.	0.	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	IFUNC							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
Z1	Volume fraction of material 1 (or color function)
UIC, VIC, WIC	Multiphase flow velocity components in the x, y, and z-directions respectively.
RHO_1	Density of fluid 1
RHO_2	Density of fluid 2
PIC	Equilibrium multifluid pressure
TIC	Equilibrium multifluid temperature
IFUNC	Option to define initial conditions using *DEFINE_FUNCTION cards:

VARIABLE	DESCRIPTION
	EQ.0: Not in use.
	EQ.1: All values for initial velocity, pressure, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x,y,z), meaning that each variable's initial profile is a function of position.

*DUALCESE_INITIAL_TWO-PHASE_SET

Purpose: Specify values to use for velocity, pressure, etc. for problem initialization of a two-phase multifluid model in an element subset of the dual CESE mesh.

Card 1	1	2	3	4	5	6	7	8
Variable	ESID	IFUNC						
Type	I	I						
Default	none	none						

Card 2	1	2	3	4	5	6	7	8
Variable	Z1	UIC	VIC	WIC	RHO_1	RHO_2	PIC	TIC
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
ESID	Element set ID (see *DUALCESE_ELEMENTSET)
IFUNC	Option to define initial conditions using *DEFINE_FUNCTION cards: EQ.0: Not in use. EQ.1: All values for initial velocity, pressure, and temperature now refer to *DEFINE_FUNCTION IDs. In these functions, the following parameters are allowed: f(x,y,z), meaning that each variable's initial profile is a function of position.
Z1	Volume fraction of material 1 (or color function)
UIC, VIC, WIC	Multiphase flow velocity components in the x, y, and z-directions, respectively

VARIABLE	DESCRIPTION
RHO_1	Density of material 1
RHO_2	Density of material 2
PIC	Equilibrium multiphase flow pressure
TIC	Equilibrium multiphase flow temperature

*DUALCESE_MAT_GAS

Purpose: Define the fluid (gas) properties in a viscous flow for the dual CESE solver. In this model, the dynamic viscosity is determined using Sutherland's formula for viscosity, and the thermal conductivity is determined using the Prandtl Number.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	C1	C2	PRND				
Type	I	F	F	F				
Default	none	1.458E-6	110.4	0.72				

VARIABLE	DESCRIPTION
MID	Material ID
C1, C2	Two coefficients in the Sutherland's formula for viscosity: $$\mu = \frac{C_1 T^{3/2}}{T + C_2} \ .$$ Here C_1 and C_2 are constants for a given gas. For example, for air at moderate temperatures $$C_1 = 1.458 \times 10^{-6}\,\mathrm{kg/msK^{1/2}}, \quad C_2 = 110.4\,\mathrm{K}$$
PRND	Prandtl Number (used to determine the coefficient of thermal conductivity). It is approximately constant for most gases. For air at standard conditions PRND = 0.72.

Remarks:

1. **Inviscid Flows.** C1 and C2 are only used to calculate the viscosity in viscous flows, so for inviscid flows, this material card is not needed.

2. **Thermal Coupling.** The Prandtl number is used to extract the thermal conductivity. It is only needed when thermal coupling with the structure is activated.

3. **Unit Consistency.** As with other solvers in LS-DYNA, you are responsible for unit consistency. For example, if dimensionless variables are used, C_1 and C_2 should be replaced by the corresponding dimensionless ones. If the dual CESE

model has a specified system of units either directly from the *DUALCESE_-MODEL card or inherited from the overall problem input, then these values need to be given in that unit system. Also, note that the formulas here require the temperature be given in either Kelvin or Rankine units.

*DUALCESE_MAT_GAS_0

Purpose: Define the fluid (gas) properties in a viscous flow for the dual CESE solver. In this model, the dynamic viscosity and thermal conductivity are specified constants.

Material Definition Cards. Include one card for each instance of this material type. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MU	K					
Type	I	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
MID	Material ID
MU	Fluid dynamic viscosity. MU = 1.81×10^{-5} kg/ms for air at 15°C.
K	Thermal conductivity of the fluid

Remarks:

1. **Fields that Depend on Problem Physics.** The viscosity is only used viscous flows, so for inviscid flows, it is not necessary to define it. The thermal conductivity is only used to calculate the heat transfer between the structure and the thermal solver when coupling is activated.

2. **Unit Consistency.** As with other solvers in LS-DYNA, you are responsible for unit consistency. For example, if dimensionless variables are used, MU should be replaced by the corresponding dimensionless one. If the dual CESE model has a specified system of units either directly from the *DUALCESE_MODEL card or inherited from the overall problem input, then these values need to be given in that unit system.

*DUALCESE_MAT_GAS_2

Purpose: Define the fluid (gas) properties in a viscous flow for the dual CESE solver. This model determines dynamic viscosity and thermal conductivity by combining Sutherland's formula with the Power law for dilute gases.

Material Definition Cards. Include one card for each instance of this material type. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MU0	SMU	K0	SK	T0		
Type	I	F	F	F	F	F		
Default	none	1.716E-5	111.	0.0241	194.0	273.0		

VARIABLE	DESCRIPTION
MID	Material ID
MU0 / SMU	Two coefficients appearing in the equation derived by combining Sutherland's formula with the Power law for dilute gases: $$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{3/2} \frac{T_0 + S_\mu}{T + S_\mu} \ .$$ Here μ is the dynamic viscosity, μ_0 is a reference value, and S_μ is an effective temperature called the Sutherland constant which is characteristic of the gas. For air at moderate temperatures, $$\mu_0 = 1.716 \times 10^{-5}\,\mathrm{Ns/m^2}\,, \qquad S_\mu = 111\ \mathrm{K}$$
K0/SK	Two coefficients appearing in the equation derived by combining Sutherland's formula with the Power law for dilute gases: $$\frac{k}{k_0} = \left(\frac{T}{T_0}\right)^{3/2} \frac{T_0 + S_k}{T + S_k} \ .$$ Here k is the thermal conductivity, k_0 is a reference value, and S_k is the Sutherland constant, which is characteristic of the gas. For air at moderate temperatures, $$k_0 = 0.0241\ \mathrm{W/m}\,, \qquad S_k = 194\ \mathrm{K}$$
T0	Reference temperature, T_0. The default value (273.0) is for air in K.

Remarks:

1. **Fields that Depend on Problem Physics.** The viscosity is only used for viscous flows, so for inviscid flows, it is not necessary to define it. The thermal conductivity is only used to calculate the heat transfer between the structure and the thermal solver when coupling is activated.

2. **Unit Consistency.** As with other solvers in LS-DYNA, you are responsible for unit consistency. For example, if dimensionless variables are used, MU should be replaced by the corresponding dimensionless one. If the dual CESE model has a specified system of units either directly from the *DUALCESE_MODEL card or inherited from the overall problem input, then these values need to be given in that unit system. Also, note that the formulas here require the temperature be given in either Kelvin or Rankine units.

*DUALCESE_MESH_GEOMETRY

Purpose: Set general purpose control variables for the dual CESE compressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	GEOM	IDC						
Type	A	F						
Default	none	0.25						
Remarks	1	2						

VARIABLE	DESCRIPTION
GEOM	Sets the geometric dimension: EQ.2D: Two-dimensional (2D) problem EQ.3D: Three-dimensional (3D) problem EQ.AXI: 2D axisymmetric
IDC	Contact interaction detection coefficient (for FSI and conjugate heat transfer problems)

Remarks:

1. **Mesh and Boundary Conditions for 2D Problems.** If you want to use the 2D (GEOM = 2D) or 2D axisymmetric (GEOM=AXI) solver, the mesh should only be distributed in the *xy*-plane with the boundary conditions given only at the *xy* domain boundaries. Otherwise, a warning message will be given, and the 3D solver will be triggered instead.

   The 2D axisymmetric case will work only if the 2D mesh and corresponding boundary conditions are properly defined with the *x* and *y* coordinates corresponding to the radial and axial directions, respectively.

2. **Contact Interaction Detection Coefficient.** IDC is the same type of variable that is input on the *ICFD_CONTROL_FSI card. For an explanation, see Remark 1 for the *ICFD_CONTROL_FSI card.

3. **Dual CESE Solver and Restarts.** The dual CESE solver is *not* currently supported for restarts.

*DUALCESE_MESH_PART

Purpose: Define dual CESE solver mesh parts, that is, specify the dual CESE FSI algorithm to use for a mesh part. In other words, this keyword allows you to restrict the type of solver (immersed boundary FSI, moving mesh FSI, or non-FSI Eulerian) used on a region of a dual CESE mesh. The part ID specified with this keyword corresponds to the second field of Card 1 for either *DUALCESE_ELE2D or *DUALCESE_ELE3D.

Part Cards. Include one card for each dual CESE part. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	FSITYPE	MMSHID					
Type	I	A	I					
Default	none	optional	o					

VARIABLE	DESCRIPTION
PID	Part ID (must be different from any other *DUALCESE_MESH_PART part ID). It is important to note that *DUALCESE_PART and *DUALCESE_PART_MULTIPHASE cards should not be used when the newer *DUALCESE_MESH_PART card is used.
FSITYPE	FSI type to use on this part: EQ.<BLANK>: If left blank, no FSI performed. EQ.IBM: Immersed boundary FSI solver EQ.MOVMESH: Moving mesh FSI solver (FSITYPE = MMM may also be used for the same effect)
MMSHID	ID for the mesh motion algorithm to use for the moving mesh FSI solver on this part (region of the current dual CESE mesh). This ID refers to the ID of an instantiation of *DUALCESE_CONTROL_MESH_MOV.

*DUALCESE_MODEL

Purpose: Set the units used by a dual CESE compressible flow problem, along with the name of the file specifying the dual CESE model. There can be any number of such models (each with a separate mesh), and each such model must be in a different file.

Card 1	1	2	3	4	5	6	7	8
Variable	UNITSYS	FILENAME						
Type	A	A						

VARIABLE	DESCRIPTION
UNITSYS	Name of the unit system of this dual CESE model (defined with *UNIT_SYSTEM). EQ.<BLANK>: Use same units as the presumed units of the entire problem.
FILENAME	Filename of the keyword file containing the dual CESE model. Note that only *DUALCESE_... keyword cards are allowed in this file.

*DUALCESE_NODE2D

Purpose: Define a node and its coordinates in the global coordinate system. The nodal point ID must be unique relative to other nodes defined with *DUALCESE_NODE2D or *DUALCESE_NODE3D cards.

Node Cards. Include as many cards in the following format as desired. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	NID	X		Y						
Type	I	F		F						
Default	none	0.		0.						

VARIABLE	DESCRIPTION
NID	Node number
X	*x* coordinate
Y	*y* coordinate

*DUALCESE_NODE3D

Purpose: Define a node and its coordinates in the global coordinate system. The nodal point ID must be unique relative to other nodes defined with *DUALCESE_NODE3D or *DUALCESE_NODE2D cards.

Node Cards. Include as many cards in the following format as desired. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	NID	X		Y		Z				
Type	I	F		F		F				
Default	none	0.		0.		0.				

VARIABLE	DESCRIPTION
NID	Node number
X	*x* coordinate
Y	*y* coordinate
Z	*z* coordinate

*DUALCESE_NODESET

Purpose: Define a nodal set of dual CESE mesh nodes.

Card 1	1	2	3	4	5	6	7	8
Variable	NSID							
Type	I							
Default	none							

Node ID Cards. Set one value per node in the set. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	NID1	NID2	NID3	NID4	NID5	NID6	NID7	NID8
Type	I	I	I	I	I	I	I	I

VARIABLE	DESCRIPTION
NSID	Set ID of new node set. All dual CESE node sets should have a unique set ID.
NID*i*	Node ID *i*

*DUALCESE_PART

Purpose: Define dual CESE solver parts, that is, specify the dual CESE material and EOS information for a part. It also provides a means to restrict the type of solver used on a region of a dual CESE mesh, meaning an immersed boundary FSI solver, a moving mesh FSI solver, or a non-FSI Eulerian solver can be specified for just this part.

> **WARNING:** As of version R14, this keyword is deprecated. Instead, for the part ID and FSI solver details, use *DUALCESE_MESH_PART. For the material properties, use the *DUALCESE_SOLVER_... card corresponding to the choice made with *DUALCESE_SOLVER_SELECTION.

Part Cards. Include one card for each dual CESE part. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	MID	EOSID	FSITYPE	MMSHID			
Type	I	I	I	A	I			
Default	none	none	none	optional	0			

VARIABLE	DESCRIPTION
PID	Part ID (must be different from any other *DUALCESE_PART, *DUALCESE_PART_MULTIPHASE, *DUALCESE_MESH_PART part ID)
MID	Material ID referring to a *DUALCESE_MAT_… material (see Remark 1)
EOSID	Equation of state ID referring to a *DUALCESE_EOS_… EOS
FSITYPE	FSI type to use on this part: EQ.<BLANK>: If left blank, no FSI performed. EQ.IBM: Immersed boundary FSI solver EQ.MOVMESH: Moving mesh FSI solver (FSITYPE = MMM may also be used for the same effect)

VARIABLE	DESCRIPTION
MMSHID	ID for the mesh motion algorithm to use for the moving mesh FSI solver on this part (region of the current dual CESE mesh). This ID refers to a *DUALCESE_CONTROL_MESH_MOV card ID.

Remarks:

1. **MID for Inviscid Flows.** Since material coefficients are only used in viscous flows, MID can be left blank for inviscid flows.

*DUALCESE_PART_MULTIPHASE

Purpose: Define dual CESE multiphase solver parts, that is, specify the dual CESE material and EOS information for a part. It also provides a means to restrict the type of solver used on a region of a dual CESE mesh, meaning an immersed boundary FSI solver, a moving mesh FSI solver, or a non-FSI Eulerian solver can be specified for just this part.

> **WARNING:** As of version R14, this keyword is deprecated. Instead, for the part ID and FSI solver details, use *DUALCESE_MESH_PART. For the material properties, use the *DUALCESE_SOLVER_... card corresponding to the choice made with *DUALCESE_SOLVER_SELECTION.

Part Cards. Include one card for each dual CESE multiphase solver part. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	REACT_ID	EOSSID	MID	FSITYPE	MMSHID		
Type	I	I	I	I	A	I		
Default	none	none	none	none	optional	0		

VARIABLE	DESCRIPTION
PID	Part ID (must be different from any PID on a *DUALCESE_PART, *DUALCESE_PART_MULTIPHASE, *DUALCESE_MESH_PART card)
REACT_ID	ID of chemical reaction rate model (see *DUALCESE_REACTION_RATE_... cards)
EOSSID	Set ID of multiphase EOS set specification (see *DUALCESE_EOS_SET)
MID	Material ID defined by a *DUALCESE_MAT_… card

VARIABLE	DESCRIPTION
FSITYPE	FSI type to use on this part: EQ.<BLANK>: If left blank, no FSI is performed. EQ.IBM: Immersed boundary FSI solver EQ.MOVMESH: Moving mesh FSI solver (FSITYPE = MMM may also be used for the same effect)
MMSHID	ID of the mesh motion algorithm to use for the moving mesh FSI solver on this part (region of the current dual CESE mesh). This ID refers to a *DUALCESE_CONTROL_MESH_MOV card ID.

*DUALCESE_POINTSET

Purpose: Define a list of points used to output at specified sample times variables from the chosen dual CESE solver to binary database binout.

Card 1	1	2	3	4	5	6	7	8
Variable	PNTSID							
Type	I							
Default	none							

Point Cards. Provide as many cards as necessary. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	X	Y	Z					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
PNTSID	ID for this point set which can be used by *DUALCESE_DATABASE_HISTORY_POINT_SET
X, Y, Z	Coordinates of a point. As many points as desired can be specified

*DUALCESE_REACTION_RATE_IG

Purpose: Define a reaction rate law for the Ignition and Growth model to describe the conversion of reactants to products in the modeling of a condensed phase explosive in a dual CESE multiphase model. See Michael and Nikiforakis 2016 and Tarver 2005 for details about this law.

Card 1	1	2	3	4	5	6	7	8
Variable	REACT_ID	IGN	AA	BB	XX	GROW1	CC	DD
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	YY	GROW2	EE	GG	ZZ	IGMAX	G1MAX	G2MAX
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
REACT_ID	ID of reaction rate law
IGN	Reaction ignition term parameter, I
AA	Reaction ignition term parameter, a
BB	Reaction ignition term parameter, b
XX	Reaction ignition term parameter, x
GROW1	Reaction growth term parameter, G_1
CC	Reaction growth term parameter, c
DD	Reaction growth term parameter, d
YY	Reaction growth term parameter, y

VARIABLE	DESCRIPTION
GROW2	Reaction completion term parameter, G_2
EE	Reaction completion term parameter, e
GG	Reaction completion term parameter, g
ZZ	Reaction completion term parameter, z
IGMAX	Maximum mass fraction of the product for reaction ignition term, Φ_{IGmax}
G1MAX	Maximum mass fraction of the product for reaction growth term, Φ_{G1max}
G2MAX	Maximum Φ for reaction completion term, Φ_{G2max}

Remarks:

Using the notation of Michael and Nikiforakis 2016, the reaction rate law can be given as:

$$\frac{d\Phi}{dt} = I(1-\Phi)^b(\rho - 1 - a)^x H(\Phi_{\text{IGmax}} - \Phi) + G_1(1-\Phi)^c\Phi^d p^y H(\Phi_{\text{G1max}} - \Phi) + G_2(1-\Phi)^e\Phi^g p^z H(\Phi - \Phi_{\text{G2max}})$$

where H is the Heaviside function. Here Φ is the mass fraction of the products, p is the pressure, and ρ is the density of the explosive mixture. Note that the pressure is assumed to be in equilibrium between the phases. I, G_1, G_2, a, b, c, d, e, g, x, y, and z are constants. They depend on the explosive as well as the part of the detonation process being modeled; see Michael and Nikiforakis 2016 for details.

The constants Φ_{IGmax}, Φ_{G1max}, and Φ_{G2max} in the Heavyside functions indicate when each of the three stages of the reaction are dominant. It was developed by Tarver 2005 to describe the reaction of pressed solid explosives during shock initiation and detonation. Tarver 2005 developed this reaction law for pressed solid explosives to describe the observed reaction stages during shock initiation and detonation See Tarver 2005 for a description of each stage of the reaction.

*DUALCESE_REACTION_RATE_IG_REDUCED

Purpose: Define a reduced form reaction rate law compared to that defined with *DUALCESE_REACTION_RATE_IG for describing the conversion of reactants to products in the modeling of a condensed phase explosive in a dual CESE multiphase model. See Michael and Nikiforakis 2016 for details about this reaction rate law.

Card 1	1	2	3	4	5	6	7	8
Variable	REACT_ID	GROW1	CC	DD	YY	PHI0		
Type	I	F	F	F	F	F		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
REACT_ID	ID of reaction rate law
GROW1	Reaction growth term parameter, G_1
CC	Reaction growth term parameter, c
DD	Reaction growth term parameter, d
YY	Reaction growth term parameter, y
PHI0	Additional parameter to account for the non-zero amount of reaction when the mass fraction of the products, ϕ, is zero

Remarks:

Michael and Nikiforakis 2016 simplified the Ignition and Growth model reaction law to a pressure dependent law with a single stage to make the reduced model. They excluded the first and third terms from the Ignition and Growth model to makes this law. However, to account for a finite amount of reaction from ignition when ϕ is zero, they added a constant ϕ_0. This reduced law has the form:

$$\frac{d\phi}{dt} = G_1(1-\phi)^c(\phi+\phi_0)^d p^y \quad .$$

All the parameters are the same as the Ignition and Growth model except Φ_0. See *DUALCESE_REACTION_RATE_IG and Michael and Nikiforakis 2016 for details.

*DUALCESE_REACTION_RATE_P_DEPEND

Purpose: Define an explicitly pressure-dependent reaction rate law for describing the conversion of reactants to products in the modeling of a condensed phase explosive in a dual CESE multiphase model. This law is from Banks et al 2008.

Card 1	1	2	3	4	5	6	7	8
Variable	REACT_ID	SIGMA	NU	N				
Type	I	F	F	F				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
REACT_ID	ID of reaction rate law
SIGMA	A positive constant, σ
NU	A positive constant, ν
N	A positive constant, n

Remarks:

Using the notation of Michael and Nikiforakis 2016, this simple pressure dependent reaction rate law by Banks et al 2008 can be stated as:

$$\frac{d\phi}{dt} = \sigma \phi^{\nu} p^{n}$$

Here ϕ is the mass fraction of the products, p is the pressure, and σ, ν, and n are positive constants found from experimental data. As discussed by Banks et al 2008, this type of reaction rate law that depends explicitly on pressure is often preferred for modeling the detonation of solid explosives because the constants can be found by fitting experimental data and other physical variables are difficult to measure.

*DUALCESE_SEGMENTSET

Purpose: Define a set of segments. For three-dimensional geometries, a segment can be triangular or quadrilateral. For two-dimensional geometries, a segment is a line defined by two nodes.

Card 1	1	2	3	4	5	6	7	8
Variable	SID							
Type	I							
Default	none							

Segment Cards. For each segment in the set include one card of this format. Include as many cards as necessary. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				

VARIABLE	DESCRIPTION
SID	Set ID. All segment sets should have a unique set ID.
N1	Nodal point n_1
N2	Nodal point n_2
N3	Nodal point n_3. To define a line segment, set N3 = N2.
N4	Nodal point n_4. To define a triangular segment, set N4 = N3. To define a line segment, set N4 = N2.

FACE	Hexahedron	Pentahedron	Pyramid	Tetrahedron
1	N1, N5, N8, N4	N1, N2, N5	N1, N4, N3, N2	N1, N2, N4
2	N2, N3, N7, N6	N4, N6, N3	N1, N2, N5	N2, N3, N4

FACE	Hexahedron	Pentahedron	Pyramid	Tetrahedron
3	N1, N2, N6, N5	N1, N4, N3, N2	N2, N3, N5	N1, N3, N2
4	N4, N8, N7, N3	N2, N3, N6, N5	N3, N4, N5	N1, N4, N3
5	N1, N4, N3, N2	N1, N5, N6, N4	N4, N1, N5	
6	N5, N6, N7, N8			

Table 5-1. Face definitions for volume dual CESE elements

*DUALCESE_SOLVER_SELECTION

Purpose: Set general purpose control variables for the dual CESE compressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EQNS							
Type	A							
Default	EULER							
Remark	1							

VARIABLE	DESCRIPTION
EQNS	Select the equations being solved with the dual CESE solver: EQ.CAVITATION: Cavitation solver EQ.EULER: Euler equations EQ.HYBRID: Hybrid multiphase model EQ.NS: Navier-Stokes equations EQ.PHASE-CHNG: Phase change model EQ.TWO-PHASE: Two-phase multiphase model

Remarks:

1. **Dual CESE Solver and Restarts.** The dual CESE solver is *not* currently supported for restarts.

*DUALCESE_SOLVER_CAV_EQNS

Purpose: Set the properties of the cavitating flow mixture being solved with the cavitation solver.

Property Card. Include one card.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
EOSID	Equation of state (EOS) ID that must refer to an EOS defined with *DUALCESE_EOS_CAV_HOMOG_EQUILIB

*DUALCESE_SOLVER_EULER_EQNS

Purpose: Set the properties of the gas flow being solved with the Euler equations solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
EOSID	Equation of state (EOS) ID that refers to an EOS defined by one of the following keywords: *DUALCESE_EOS_IDEAL_GAS, *DUALCESE_EOS_INFLATOR1, or *DUALCESE_EOS_INFLATOR2.

*DUALCESE_SOLVER_HYBRID_MULTIPHASE

Purpose: Define the properties of the fluid and condensed phase materials via equations-of-state that are used together to compute the thermodynamic state in the hybrid multi-phase dual CESE solver. Also, specify the single-step reaction rate model for the chemical reactions.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSINID	EOSRCTID	EOSPRDID	REACT_ID				
Type	I	I	I	I				
Default	none	none	none	none				
Remark	1	1	1	2				

VARIABLE	DESCRIPTION
EOSINID	EOS ID of the inert component of the multiphase mixture
EOSRCTID	EOS ID of the reactant phase of the multiphase mixture
EOSPRDID	EOS ID of the product phase of the multiphase mixture
REACT_ID	ID of chemical reaction rate model

Remarks:

1. **Supported EOS Types.** Each EOS can be one of the following types:

   *DUALCESE_EOS_IDEAL_GAS

   *DUALCESE_EOS_STIFFENED_GAS

   *DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED

   *DUALCESE_EOS_COCHRAN_CHAN

   *DUALCESE_EOS_JWL.

2. **Supported Reaction Rate Types.** The reaction rate can be one of the following types:

*DUALCESE_REACTION_REAT_IG

*DUALCESE_REACTION_RATE_IG_REDUCED

*DUALCESE_REACTION_RATE_P_DEPEND

*DUALCESE_SOLVER_NAVIER_STOKES

Purpose: Set the properties of the fluid flow being solved with the Navier-Stokes equations solver. That is, specify the dual CESE material and EOS information to be used.

Property Card. Include one card.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	MID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
EOSID	Equation of state (EOS) ID referring to an EOS defined by one of the following card types: *DUALCESE_EOS_IDEAL_GAS, *DUALCESE_EOS_INFLATOR1, or *DUALCESE_EOS_INFLATOR2.
MID	Material ID referring to a *DUALCESE_MAT_… material card (see Remark 1)

Remarks:

1. **MID for Inviscid Flows.** Since material coefficients are only used in viscous flows, MID can be left blank for inviscid flows.

*DUALCESE_SOLVER_PHASE_CHANGE

Purpose: Define the properties of the fluid and condensed phase materials via equations of state that are used together to compute the thermodynamic state in the phase change multiphase dual CESE solver.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSIDL	EOSIDV	EOSIDO					
Type	I	I	I					
Default	none	none	none					
Remarks	1	1	1					

VARIABLE	DESCRIPTION
EOSIDL	Equation of state (EOS) ID of the liquid component in the multiphase mixture
EOSIDV	Equation of state (EOS) ID of the vapor component in the multiphase mixture
EOSIDO	Equation of state (EOS) ID of the other gases in the multiphase mixture

Remarks:

1. **Available EOS models.** Each EOS can be one of the following types:

   *DUALCESE_EOS_NASG

*DUALCESE_SOLVER_TWO-PHASE_MULTIPHASE

Purpose: Define the properties of the fluid phase materials through equations-of-state that are used together to compute the thermodynamic state in the two-phase multiphase dual CESE solver. This solver does not compute phase changes.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID1	EOSID2						
Type	I	I						
Default	none	none						
Remarks	1	1						

VARIABLE	DESCRIPTION
EOSID1	EOS ID of the first inert component of the multiphase mixture
EOSID2	EOS ID of the second inert component of the multiphase mixture

Remarks:

1. **Supported EOS Types.** Each EOS can be one of the following types:

   *DUALCESE_EOS_IDEAL_GAS

   *DUALCESE_EOS_STIFFENED_GAS

   *DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED

   *DUALCESE_EOS_COCHRAN_CHAN

   *DUALCESE_EOS_JWL

*EM

The *EM keyword cards provide input for the electromagnetism module. This module is for solving 3D eddy-current, inductive heating or resistive heating problems. It can be coupled with the mechanical and thermal solvers. Typical applications include magnetic metal forming and welding. This module also includes coupling the boundary element method to the finite element method coupling. We intend this coupling for simulations involving a conductor interacting with air so that the air does not need to be meshed. The conductor is modeled with finite elements while the air is modeled with boundary elements.

*EM_2DAXI

*EM_BOUNDARY

*EM_BOUNDARY_PRESCRIBED

*EM_CIRCUIT

*EM_CIRCUIT_CONNECT

*EM_CIRCUIT_ROGO

*EM_CONTACT

*EM_CONTACT_RESISTANCE

*EM_CONTACT_SUBDOM

*EM_CONTROL

*EM_CONTROL_CONTACT

*EM_CONTROL_COUPLING

*EM_CONTROL_EROSION

*EM_CONTROL_MAGNET

*EM_CONTROL_SOLUTION

*EM_CONTROL_SWITCH

*EM_CONTROL_SWITCH_CONTACT

*EM_CONTROL_TIMESTEP

*EM_DATABASE_CIRCUIT

*EM_DATABASE_CIRCUIT0D

*EM_DATABASE_ELOUT

*EM_DATABASE_FIELDLINE

*EM_DATABASE_GLOBALENERGY

*EM_DATABASE_NODOUT

*EM_DATABASE_PARTDATA

*EM_DATABASE_POINTOUT

*EM_DATABASE_ROGO

*EM_DATABASE_TIMESTEP

*EM_EP_CELLMODEL_DEFINEFUNCTION

*EM_EP_CELLMODEL_FENTONKARMA

*EM_EP_CELLMODEL_FIZHUGHNAGUMO

*EM_EP_CELLMODEL_TENTUSSCHER

*EM_EP_CELLMODEL_TOMEK

*EM_EP_CELLMODEL_TOR_ORD

*EM_EP_CELLMODEL_USERMAT

*EM_EP_CREATEFIBERORIENTATION

*EM_EP_ECG

*EM_EP_ISOCH

EM_EP_LAPLACE_DIRICHLET

*EM_EP_PURKINJE_NETWORK

*EM_EP_TENTUSSCHER_STIMULUS

*EM_EOS_BURGESS

*EM_EOS_MEADON

*EM_EOS_PERMEABILITY

*EM_EOS_TABULATED1

*EM_EOS_TABULATED2

*EM_EXTERNAL_FIELD

*EM_ISOPOTENTIAL

*EM_ISOPOTENTIAL_CONNECT

*EM_ISOPOTENTIAL_ROGO

*EM_MAT_001

*EM_MAT_002

*EM_MAT_003

*EM_MAT_004

*EM_MAT_005

*EM_MAT_006

*EM_OUTPUT

*EM_OUTPUT_VTK

*EM_PERMANENT_MAGNET

*EM_POINT_SET

*EM_RANDLES_BATMAC

*EM_RANDLES_EXOTHERMIC_REACTION

*EM_RANDLES_MESHLESS

*EM_RANDLES_TSHELL

*EM_RANDLES_SHORT

*EM_RANDLES_SOLID

*EM_ROTATION_AXIS

*EM_SOLVER_BEM

*EM_SOLVER_BEMMAT

*EM_SOLVER_FEM

*EM_SOLVER_FEMBEM

*EM_SOLVER_FEMBEM_MONOLITHIC

*EM_2DAXI

Purpose: Sets up the electromagnetism solver as 2D axisymmetric instead of 3D, on a given part, in order to save computational time as well as memory.

The electromagnetism is solved in 2D on a given cross section of the part (defined by a segment set), with a symmetry axis defined by its direction (at this time, it can be the x, y, or z axis). The EM forces and Joule heating are then computed over the full 3D part by rotations. The part needs to be compatible with the symmetry, i.e. each node in the part needs to be the child of a parent node on the segment set, by a rotation around the axis. Only the conductor parts (with a *EM_MAT_… of type 2 or 4) should be defined as 2D axisymmetric.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	SSID			STARSSID	ENDSSID	NUMSEC	
Type	I	I			I	I	I	
Default	none	none			none	none	none	

VARIABLE	DESCRIPTION
PID	Part ID of the part to be solved using 2D axisymmetry
SSID	Segment Set ID : Segment that will define the 2D cross section of the part where the EM field is solved
STARSSID, ENDSSID	Used by the 2D axisymmetric solver to make the connection between two corresponding boundaries on each side of a slice when the model is a slice of the full 360 circle.
NUMSEC	Number of Sectors. This field gives the ratio of the full circle to the angular extension of the mesh. This has to be a power of two. For example, NUMSEC = 4 means that the mesh of the part represents one fourth of the total circle. If this value is set to 0, then the value from *EM_ROTATION_AXIS is used instead.

Remarks:

1. At this time, *either* all or none of the conductor parts should be 2D axisymmetric. In the future, a mix between 2D axisymmetric and 3D parts will be allowed.

*EM_BOUNDARY

Purpose: Define some boundary conditions for the electromagnetism problems.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	BTYPE						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
SSID	Segment Set Id
BTYPE	EQ.9: The faces of this segment set are eliminated from the BEM calculations (used for example for the rear or side faces of a workpiece).

*EM_BOUNDARY_PRESCRIBED

Purpose: Prescribe a local boundary condition applied on nodes.

Card 1	1	2	3	4	5	6	7	8
Variable	BPID	BPTYPE	SETTYPE	SETID	VAL	LCID		SYSTYPE
Type	I	I	I	I	F	I		I
Default	none	none	none	none	0.	0		0

Optional Card.

Card 2	1	2	3	4	5	6	7	8
Variable	BIRTHT	DEATHT						
Type	F	F						
Default	0.	1.e28						

VARIABLE	DESCRIPTION
ISOID	ID of the prescribed boundary
BPTYPE	Prescribed boundary type: EQ.1: Short (scalar potential set to 0.) EQ.2: Prescribed resistance (Robin B.C.) EQ.3: Prescribed scalar potential (Dirichlet B.C.) EQ.4: Prescribed current density (Neumann B.C.)
SETTYPE	Set type: EQ.1: Segment set EQ.2: Node set EQ.3: Fluid part (see *ICFD_PART)
SETID	Set ID

VARIABLE	DESCRIPTION
VAL	Value of the resistance, current density or potential depending on BPTYPE. Ignored if LCID is defined.
LCID	Load curve ID defining the value of the resistance, voltage, or current as a function of time. If a negative value is entered, a *DEFINE_FUNCTION will be expected. The following parameters are allowed: f(time, emdt, curr, pot, cond, temp, potglob, currglob, areaglob, area, x, y, z). Pot/curr/area and potglob/currglob/areaglob are the local value of the scalar potential/current/area and the global averaged value on the prescribed boundary, respectfully. Cond is the local electrical conductivity, and x, y, and z are the local coordinates.
SYSTYPE	Flag for the type of system on which the boundary condition is applied (applies only for cardiac electrophysiology when *EM_BOUNDARY_PRESCRIBED is used with EMSOL = 11 or 12): EQ.0: Applied on extracellular potential EQ.1: Applied on transmembrane potential
BIRTHT / DEATHT	Birth and death times for that prescribed boundary

Remarks:

1. **Supported Solvers.** This keyword is currently only available for the resistive heating solver (EMSOL = 3) and the electrophysiology solvers (EMSOL = 11, 12, and 13).

*EM_CIRCUIT_{OPTION}

Available options include

SOURCE

Purpose: Define an electrical circuit.

For the SOURCE option, the current will be considered uniform in the circuit. In general, this feature is used to model stranded conductors carrying a source current (in which case Amperes become Ampere-turns). This feature can also be useful for saving computational time in cases with a low frequency current and where the diffusion of the EM fields is a very fast process. In contrast, for the general case, the current density in a circuit is modeled in accordance with the solver type defined in EMSOL of *EM_CONTROL. For example, if an eddy current solver is selected, the diffusion of the current in the circuit is taken into account. See Remark 4 for a discussion of the available circuit types for source circuits.

Card 1	1	2	3	4	5	6	7	8
Variable	CIRCID	CIRCTYP	LCID	R/F	L/A	C/to	V0	T0
Type	I	I	I	F	F	F	F	F
Default	none	none	none	none	none	none	none	0.

Card 2	1	2	3	4	5	6	7	8
Variable	SIDCURR	SIDVIN	SIDVOUT	PARTID	IFREQST			
Type	I	I	I	I	I			
Default	none	none	none	none	1			

VARIABLE	DESCRIPTION
CIRCID	Circuit ID
CIRCTYP	Circuit type (see Remark 4 for source circuits): EQ.1: Imposed current vs time defined by a load curve.

VARIABLE	DESCRIPTION
	EQ.2: Imposed voltage vs time defined by a load curve. If a negative value is entered for LCID, its absolute value will refer to a DEFINE FUNCTION for a user defined circuit equation. If a DEFINE_FUNCTION is used, the following parameters are accepted: f(time, emdt, curr, curr1, curr2, pot1, pot2). emdt is the current timestep; curr, curr1, and curr2 refer to the current value at t, $t-1$ and $t-2$, respectfully; and pot1 and pot2 refer to the scalar potential at $t-1$ and $t-2$, respectfully.
	EQ.3: R, L, C, V0 circuit (not available for source circuits)
	EQ.11: Imposed current defined by an amplitude A, frequency F and initial time t_0: $I = A\sin[2\pi F(t-t_0)]$
	EQ.12: Imposed voltage defined by an amplitude A, frequency F and initial time t_0: $V = A\sin[2\pi F(t-t_0)]$
	EQ.21: Imposed current defined by a load curve over one period and a frequency F. See Remark 3.
	EQ.22: Imposed voltage defined by a load curve over one period and a frequency F. See Remark 3.
LCID	Load curve ID for CIRCTYP = 1, 2, 21 or 22
R/F	Value of the circuit resistance for CIRCTYP = 3 Value of the frequency for CIRCTYP = 11, 12, 21 or 22. For CIRCTYP = 11 or 12, to have the frequency specified as a function of time with a load curve, a negative value can be entered with the absolute value corresponding to the load curve ID.
L/A	Value of the circuit inductance for CIRCTYP = 3 Value of the amplitude for CIRCTYP = 11 or 12. To have the amplitude specified as a function of time with a load curve, a negative value can be entered with the absolute value corresponding to the load curve ID.
C/t0	Value of the circuit capacity for CIRCTYP = 3 Value of the initial time t_0 for CIRCTYP = 11 or 12
V0	Value of the circuit initial voltage for CIRCTYP = 3.
T0	Starting time for CIRCTYPE = 3. Default is at the beginning of the run.

VARIABLE	DESCRIPTION
SIDCURR	Segment set ID for the current. It uses the orientation given by the normal of the segments. To use the opposite orientation, use a '–' (minus) sign in front of the segment set ID. CIRCTYP.EQ.1/11/21: The current is imposed through this segment set. CIRCTYP.EQ.3: The current needed by the circuit equations is measured through this segment set. CIRCTYP.EQ.2/12/22: Optional segment set that the current is forced through. See Remark 2.
SIDVIN	Segment set ID for input voltage when CIRCTYP = 2, 3, 12, or 22 or for input current when CIRCTYP = 1, 11, or 21. The input voltage or current is oriented to enter the structural mesh, irrespective of the orientation of the segments.
SIDVOUT	Segment set ID for output voltage when CIRCTYP = 2, 3, 12, or 22 or for output current when CIRCTYP = 1, 11, or 21. The output voltage or current is oriented to leave the structural mesh, irrespective of the orientation of the segment.
PARTID	Part ID associated to the circuit. It can be any part ID associated to the circuit.
IFREQST	Frequency for recomputing the source terms for the SOURCE keyword option only. The source terms are recalculated every IFREQST time steps. By default, the source terms are recomputed every EM time step. LT.0: |IFREQST| is a load curve ID giving the frequency for recomputing as a function of time.

Remarks:

1. **Imposed Current with Closed Loop Geometry.** When defining a circuit with an imposed current (CIRCTYP 1, 11 or 21) in cases of a closed loop geometry (torus), SIDVIN and SIDVOUT cannot be defined and thus, only SIDCURR is necessary.

2. **SIDCURR with Imposed Tension.** When defining a circuit with an imposed tension (CIRCTYP 2, 12, 22), it is possible to also define SIDCURR. This can be

Variable	Circuit Type (CIRCTYP)				
	Im-posed 1: Current	Imposed 2: Voltage	3: R, L, C	11: F, A, t0	12: F, A, t0
LCID	M	M	-	-	-
R/L/C/V0	-	-	M	-	-
F	-	-	-	M	M
A/t0	-	-	-	M	M
SIDCURR	M	O	M	M	O
SIDVIN	M*	M	M	M*	M
SIDVOUT	M*	M	M	M*	M
PARTID	M	M	M	M	M
Variable	21: LCID, F	22 : LCID, F			
LCID	M	M	-	-	-
R/L/C/V0	-	-	-	-	-
F	M	M	-	-	-
A/t0	-	-	-	-	-
SIDCURR	M	O	-	-	-
SIDVIN	M*	M	-	-	-
SIDVOUT	M*	M	-	-	-
PARTID	M	M	-	-	-

Table 6-1. Correspondence between circuit type and card entries. "M" indicates mandatory, "M*" mandatory with exceptions (see Remark 1), "O" indicates optional, and "-" indicates ignored.

useful in circuits where various flow paths are possible for the current in order to force the entire current to go through SIDCURR.

3. **CIRCTYP = 21 and 22.** Circuit types 21 and 22 are for cases where the periodic current/tension does not exactly follow a perfect sinusoidal. You must provide the shape of the current/tension over one period through a LCID as well as the frequency.

4. **Circuit Types for Source Circuits.** For source circuits, CIRCTYP = 1, 2, 11, 12, 21, and 22 are supported. For imposed currents (CIRCTYP = 1, 11, or 21), source

circuits must be associated with an insulator material (MTYPE = 1) in *EM_-MAT_001. The solver relies on the Biot-Savart integration method to retrieve the influence of the source circuit on other conductors. This method is fast, but it is not possible to calculate the force on the coil.

You can also define a source circuit with an imposed voltage with CIRCTYP = 2, 12, or 22 (associated to a resistance value and a number of windings). The source circuit must be used with the monolithic solver (*EM_SOLVER_FEMBEM_-MONOLITHIC) and become part of the FEM/BEM system. Therefore, it must be associated with a conductor material (MTYPE = 2) on *EM_MAT_001. The imposed voltage is slower than the imposed current, but a force applied to the coil can be calculated. Defining source circuits by their voltage instead of their current is very useful in cases where the current is not known, such as for actuators or electric motors.

*EM_CIRCUIT_CONNECT

Purpose: This keyword connects several circuits together by imposing a linear constraint on the global currents of circuit pairs

$$c_1 i_1 + c_2 i_2 = 0.$$

This is especially useful for 2D axisymmetric models involving spiral or helical coils.

Card 1	1	2	3	4	5	6	7	8
Variable	CONID	CONTYPE	CIRC1	CIRC2	C1	C2		
Type	I	I	I	I	F	F		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
CONID	Id of the Circuit Connect
CONTYPE	Type of connection between circuits. For the moment, it is only possible to combine circuits by imposing a linear constraint on the global current (=1).
C1/C2	Values of the linear constraints if CONTYPE = 1.

*EM_CIRCUIT_ROGO

Purpose: Define Rogowsky coils to measure a global current vs time through a segment set or a node set.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	ROGID	SETID	SETTYPE	CURTYP				
Type	I	I	I	I				
Default	0	0	0	0				

VARIABLE	DESCRIPTION
ROGID	Rogowsky coil ID
SETID	Segment or node set ID
SETTYPE	Type of set: EQ.1: Segment set EQ.2: Node set (not available yet}
CURTYP	Type of current measured: EQ.1: Volume current EQ.2: Surface current (not available yet} EQ.3: Magnetic field flow (B field times Area)

Remarks:

1. An ASCII file "em_rogo_*xxx*" , with *xxx* representing the rogoId, is generated for each *EM_CIRCUIT_ROGO card giving the value of the current or the magnetic field vs time.

*EM_CONTACT

Purpose: Optional define and specify options on electromagnetic contacts between two sets of parts. Generally, it is used with the *EM_CONTACT_RESISTANCE. Fields left empty on this card default to the value of the equivalent field for *EM_CONTROL_CONTACT.

Contact Definition Cards. Include one card for each contact definition. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	CONTID	DTYPE	PSIDR	PSIDT	EPS1	EPS2	EPS3	D0
Type	I	I	I	I	F	F	F	F
Default	none	0	none	none	0.3	0.3	0.3	none

VARIABLE	DESCRIPTION
CONTID	Electromagnetic contact ID
DTYPE	Detection type (see remarks): EQ.0: Contact type 0 (default) EQ.1: Contact type 1
PSIDR	Reference surface part set ID
PSIDT	Tracked surface part set ID
EPS*i*	Contact coefficients for contact detection conditions. See discussion below.
D0	Contact condition 3 when COTYPE = 1 (see remarks)

Remarks:

In these remarks we will discuss the conditions for contact detection. For reference, Figure 0-1 illustrates which geometric values help determine contact. In this figure and discussion, f_1 is a face of the reference surface, and f_2 is a face of the tracked surface. Contact is detected when *all of the following three condition are satisfied*:

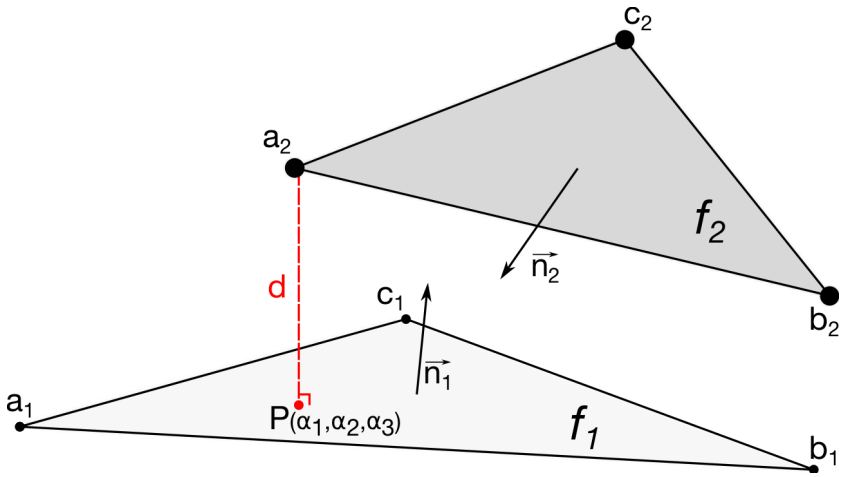


Figure 0-1. Contact detection conditions between two faces.

1. Contact condition 1:

$$\mathbf{n}_1 \cdot \mathbf{n}_2 \leq -1 + \varepsilon_1$$

Here $\mathbf{n}_1$ and $\mathbf{n}_2$ are the normal vectors of faces f_1 and f_2, respectfully, and ε_1 is an input parameter. See Figure 0-1.

2. Contact condition 2:

$$-\varepsilon_2 \leq \alpha_1 \leq 1 + \varepsilon_2$$
$$-\varepsilon_2 \leq \alpha_2 \leq 1 + \varepsilon_2$$
$$-\varepsilon_2 \leq \alpha_3 \leq 1 + \varepsilon_2$$

$(\alpha_1, \alpha_2, \alpha_3)$ are the local coordinates of point P (see Figure 0-1). P is the projection of point a_2 on face f_1. ε_2 is an input parameter.

3. Contact condition 3 depends on the contact type. For either possible condition, let d be the distance between P and a_2 (see Figure 0-1).

   a) For contact type 0:

$$d \leq \varepsilon_3 S_1 \ ,$$

   where ε_3 is an input parameter and S_1 is the minimum side length for f_1:

$$S_1 = \min[d(a_1, b_1), d(b_1, c_1), d(c_1, a_1)] \ .$$

   b) For contact type 1:

$$d \leq D_0 \ ,$$

   where D_0 is an input parameter.

*EM_CONTACT_RESISTANCE

Purpose: Calculate the electric contact resistance of a previously defined EM contact in *EM_CONTACT.

Card 1	1	2	3	4	5	6	7	8
Variable	CRID	CONTID	CTYPE		JHRTYPE			
Type	I	I	I		I			
Default	none	none	none		none			

Cards 2	1	2	3	4	5	6	7	8
Variable	DFID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
CRID	Resistive contact ID
CONTID	EM contact ID defined in *EM_CONTACT
CTYPE	Contact resistance type: EQ.1: Electric contact resistance defined by user defined define function.
DFID	Define function ID (see Remark 1)
JHRTYPE	Indicates how the Joule heating calculated by the contact resistance shall be taken into account: EQ.0: No addition. The Joule heating calculated by the contact resistance is not taken into account.

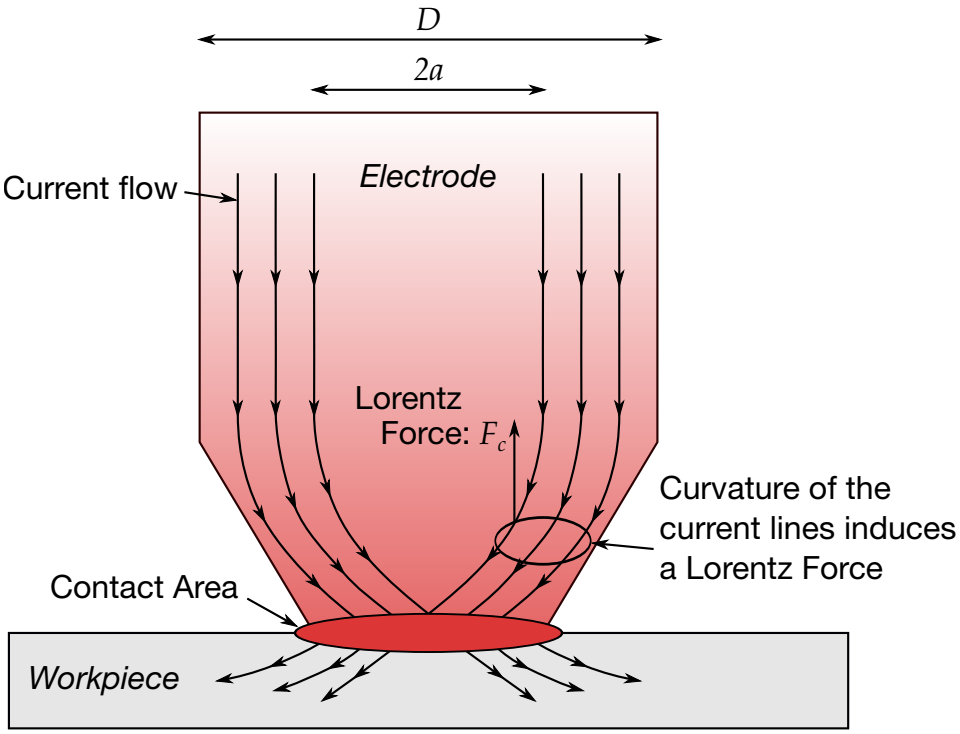


Figure 6-2. Electrode coming into contact with workpiece (RSW application).

VARIABLE	DESCRIPTION
	EQ.1: The Joule heating coming from the contact resistance is divided and distributed evenly among all elements neighboring the contact surface.

Remarks:

1. **Define Function Parameters.** In the *DEFINE_FUNCTION, the following parameters are allowed: f(time, emdt, arealoc, areatot, ctdist, currloc, currglob, rho_mst, rho_slv, cond_mst, cond_slv, ctpress_mst, ctpress_slv, temp_mst, temp_slv, vmstress_mst, vmstress_slv, press_mst, press_slv). Here,

time	current time step
emdt	EM time step
arealoc	Local area associated with each face in contact
areatot	Total contact area
ctdist	Contact distance between the two faces in contact
currloc	Local current density at the contact area
currglob	Total current flowing through the contact area

rho_mst	Density of the elements associated with the reference surface side of the contact
rho_slv	Density of the elements associated with the tracked surface side of the contact
cond_mst	Electrical conductivity of the elements associated with the reference surface side of the contact
cond_slv	Electrical conductivity of the elements associated with the tracked surface side of the contact
ctpress_mst	Contact pressure of the elements associated with the reference surface side of the contact
ctpress_slv	Contact pressure of the elements associated with the tracked surface side of the contact
temp_mst	Temperature of the elements associated with the reference surface side of the contact
temp_slv	Temperature of the elements associated with the tracked surface side of the contact
vmstress_mst	von Mises stress of the elements associated with the reference surface side of the contact
vmstress_slv	von Mises stress of the elements associated with the tracked surface side of the contact
press_mst	Pressure of the elements associated with the reference surface side of the contact
press_slv	Pressure of the elements associated with the tracked surface side of the contact

*EM_CONTACT_SUBDOM

Purpose: Optional card used for defining a specific region where EM contact will be active. This allows saving some calculation time by limiting the contact search area. Must be used in conjunction with *EM_CONTROL_CONTACT.

Card 1	1	2	3	4	5	6	7	8
Variable	SDTYPE	MVTYPE	LCIDX/NID	LCIDY	LCIDZ			
Type	I	I	I	I	I			
Default	none	0	none	none	none			

Card 2	1	2	3	4	5	6	7	8
Variable	R	PMINX	PMINY	PMINZ	PMAXX	PMAXY	PMAXZ	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
SDTYPE	Subdomain definition type : EQ.1: Defined by box. EQ.2: Defined by cylinder. EQ.3: Defined by sphere.
MVTYPE	Movement type of subdomain : EQ.0: Static subdomain (Default). EQ.1: Domain translates in the three directions by the velocities given by LCIDX,LCIDY,LCIDZ. EQ.2: Domain follows the displacements of the node ID given by NID.

VARIABLE	DESCRIPTION
LCIDX/NID	Time dependent load curve ID for the translational velocity in the X direction for MVTYPE = 1, Node ID for MVTYPE = 2.
LCIDY/Z	Time dependent load curve IDs for MVTYPE = 1 in the Y and Z directions.
R	Radius of the sphere if SDTYPE = 3 or the cylinder if SDTYPE = 2.
PMINX/Y/Z	Point of minimum coordinates if SDTYPE = 1. Origin point if SDTYPE = 3. Axis head point if SDTYPE = 2.
PMAXX/Y/Z	Point of maximum coordinates if SDTYPE = 1. Axis tail point if SDTYPE = 2.

*EM_CONTROL

Purpose: Enable the EM solver and set its options.

Card 1	1	2	3	4	5	6	7	8
Variable	EMSOL	NUMLS	MACRODT	DIMTYPE	NPERIO		NCYLFEM	NCYLBEM
Type	I	I	F	I	I		I	I
Default	0	100	none	0	2		5000	5000

VARIABLE	DESCRIPTION
EMSOL	Electromagnetism solver selector: EQ.-1: Turns the EM solver off after reading the EM keywords. EQ.1: Eddy current solver EQ.2: Induced heating solver EQ.3: Resistive heating solver EQ.11: Electrophysiology monodomain EQ.12: Electrophysiology bidomain EQ.13: Electrophysiology monodomain coupled with bidomain
NUMLS	Number of local EM steps in a whole period for EMSOL = 2. If a negative value is entered, it will give the number of local EM steps as a function of the macro time.
MACRODT	Macro time step when EMSOL = 2
DIMTYPE	EM dimension type: EQ.0: 3D solve EQ.1: 2D planar with zero thickness shell elements EQ.3: 2D axisymmetric (Y-axis only) with zero thickness elements

VARIABLE	DESCRIPTION
NPERIO	Number of periods for which the last is used to calculate the average Joule heat rate when EMSOL = 2. NPERIO = 2 means that two periods of NUMLS steps will be calculated. Only the last period of NPERIO is used for the average Joule heat calculation. See Remark 1.
NCYLFEM	Number of electromagnetism cycles between the recalculation of FEM matrices. If a negative value is entered, then the absolute value refers to a load curve giving the number of electromagnetism cycles as a function of time.
NCYLBEM	Number of electromagnetism cycles between the recalculation of BEM matrices. If a negative value is entered, then the absolute value refers to a load curve giving the number of electromagnetism cycles as a function of time.

Remarks:

1. **Number of Periods and the Average Joule Heating Calculation.** The purpose of using more than one period to calculate the average Joule heating is to allow the different fields to adopt the correct amplitude and time shift when starting from 0.0 at $t = 0$. NPERIO = 2 means that two periods will be calculated of which only last one, being the second one in this case, will be used for the average Joule heat calculation. In some cases, using higher values might be required to achieve good accuracy and conversely, in others, NPERIO = 1 might yield sufficient accuracy.

*EM_CONTROL_CONTACT

Purpose: This keyword activates the electromagnetism contact algorithms, which detects contact between conductors. Electromagnetic fields flow from one conductor to another when detected as in contact.

Card 1	1	2	3	4	5	6	7	8
Variable	EMCT	CCONLY	CTYPE	DTYPE	EPS1	EPS2	EPS3	D0
Type	I	I	I	I	F	F	F	F
Default	0	0	0	0	0.3	0.3	0.3	none

VARIABLE	DESCRIPTION
EMCT	EM contact activation flag: EQ.0: No contact detection EQ.1: Contact detection
CCONLY	Determines on which parts of the model the EM contact should be activated. EQ.0: Contact detection between all active parts associated with a conducting material. (Default) EQ.1: Only look for EM contact between parts associated through the EM_CONTACT card. In some cases this option can reduce the calculation time.
CTYPE	Contact type : EQ.-1: Node to node contact based on constraints on the scalar potential. See Remark 1. EQ.0: Node to node penalty based contact on the scalar potential. EQ.1: Discrete mortar penalty contact on the scalar potential. EQ.2: Continuous mortar penalty contact on the scalar potential and the vector potential (when active).

VARIABLE	DESCRIPTION
DTYPE	Detection type. If *EM_CONTACT is not defined, the solver will look for global contact options in *EM_CONTROL_CONTACT. EQ.0: Contact type 0 (Default). See *EM_CONTACT. EQ.1: Contact type 1.
EPS*i*	Global contact coefficients used if the equivalent fields in *EM_-CONTACT are empty.
D0	Global contact condition 3 value when DTYPE = 1

Remarks:

1. In versions prior to R12, CTYPE = 0 was the default EM contact for the resistive heating solver while CTYPE = -1 was the default EM contact for the Eddy current solver. CTYPE = 1 and CYPE = 2 are the recommended contacts for best accuracy.

2. When the Eddy current solver is active, when contact occurs between BEM surfaces, the solver will automatically remove the faces that are on the contact surface and internally stitch the two BEM surfaces together in order to achieve a continuous closed BEM mesh.

*EM_CONTROL_COUPLING

Purpose: Control couplings between various solvers with the EM solver.

Card 1	1	2	3	4	5	6	7	8
Variable	THCPL	SMCPL	THLCID	SMLCID	THCPLFL	SMCPLFL		
Type	I	I	I	I	I	I		
Default	0	0	0	0	0	0		

Optional Card.

Card 2	1	2	3	4	5	6	7	8
Variable	SMMOD	DFX	DFY	DFZ				
Type	I	I	I	I				
Default	0	none	none	none				

VARIABLE	DESCRIPTION
THCPL	Coupling to the thermal solver. When turned on, the EM solver will transfer the Joule heating terms to the solid mechanics thermal solver. EQ.0: Coupling on. EQ.1: Coupling off.
SMCPL	Coupling to the solid mechanics solver. When turned on, the EM solver will transfer forces to the solid mechanics solver. EQ.0: Coupling on. Lorentz force density is transferred. EQ.1: Coupling off. EQ.2: Coupling on. Magnetic force surface density is transferred. More accurate representation of EM forces in cases involving magnets or nonlinear ferromagnets. See *EM_SOLVER_FEMBEM_MONOLITHIC.

VARIABLE	DESCRIPTION
	EQ.3: Coupling on. Magnetic force surface density is transferred to magnets and ferromagnets while Lorentz force density is transferred to regular conductors.
THLCID	Optional load curve ID. When defined, the heat rate transferred to the thermal solver will be scaled by the value returned by THLCID.
SMLCID	Optional load curve ID. When defined, the forces transferred to the solid mechanics solver will be scaled by the value returned by SMLCID.
THCPLFL	Coupling to the heat equation when EM quantities are solved on fluid elements. When turned on, the EM solver will transfer the Joule heating terms to the ICFD solver. EQ.0: Coupling off. EQ.1: Coupling on.
SMCPLFL	Interaction between the solid mechanics solver and the ICFD solver when EM quantities are solved on fluid elements. EQ.0: The fluid pressure will be passed to the solid mechanics solver (default). EQ.1: The fluid pressure is replaced by the electrostatic pressure. EQ.2: Both the fluid pressure and the electrostatic pressure are passed on to the solid mechanics solver.
SMMOD	Coupling to the solid mechanics solver. When turned on, the EM solver will transfer forces to the solid mechanics solver. EQ.0: Off. EQ.1: Force calculation at element level is decided by *DEFINE_FUNCTION. See DFX, DFY and DFZ. EQ.2: Force calculation at element level is decided by usermat routine. See dyn21em.f and `user_getEMForceArray` routine.
DFX/DFY/DFZ	Define function IDs for the force three components if SMMOD = 1. Arguments for the define functions are the same as in *EM_EOS_TABULATED2.

*EM_CONTROL_EROSION

Purpose: Allows the EM solver to take eroded elements into account

Card 1	1	2	3	4	5	6	7	8
Variable	ECTRL							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
ECTRL	Erosion search : EQ.0: Off. This means that the EM solver will ignore eroded elements and still consider them part of the EM problem. EQ.1: On. The EM solver will look for potential elements that are eroded and remove them from the EM solve by updating its matrix system.

*EM_CONTROL_MAGNET

Purpose: Control how often the magnetization vector for magnets is recomputed. By default, the magnetization vector for each magnet is computed only once at the beginning of the calculation (see Remark 1).

Card 1	1	2	3	4	5	6	7	8
Variable	MCOMP	NCYCM						
Type	I	I						
Default	0	0						

VARIABLE	DESCRIPTION
MCOMP	Flag controlling whether the magnetization vector is recomputed. EQ.0: Magnetization vector not recomputed (see Remark 1). EQ.1: Magnetization vector recomputed at the frequency controlled by NCYCM.
NCYCM	Magnetization vector recomputation frequency. A value of 1 means recomputation at every EM time step. If a negative value is entered, |NCYCM| is the ID of a load curve giving the value as a function of time.

Remark:

1. **MCOMP.** In most applications involving magnets, the magnetization vector associated to each magnet needs only to be calculated once at the beginning of the analysis. The magnetization direction is then scaled by the Coercive force value (See EM_PERMANENT_MAGNET) to correctly estimate the contribution of each magnet. In certain specific applications, for examples in cases involving magnet deformations or changes in magnet boundary conditions, it may be needed to periodically update the initially computed magnetization vector. This can be achieved by setting MCOMP to 1 and defining NCYCM.

*EM_CONTROL_SOLUTION

Purpose: Allows the user to specify different conditions under which the FEM and BEM matrices are reassembled.

Card 1	1	2	3	4	5	6	7	8
Variable	NCYLFEM	NCYLBEM	AUTOFEM	AUTOBEM	TOL1FEM	TOL2FEM	TOL1BEM	TOL2BEM
Type	I	I	I	I	F	F	F	F
Default	5000	5000	0	0	0.3	0.1	0.3	0.1

VARIABLE	DESCRIPTION
NCYLFEM	Number of electromagnetism cycles between the recalculation of FEM matrices. If a negative value is entered, then the absolute value will refer to a load curve giving NCYCLFEM function of time.
NCYLBEM	Number of electromagnetism cycles between the recalculation of BEM matrices. If a negative value is entered, then the absolute value will refer to a load curve giving NCYCLBEM function of time.
AUTOFEM	In addition to NCYLFEM, this triggers an automatic recomputation of the FEM matrices based on an error calculation of the conductors' relative deformation and electrical conductivity changes. See TOL1FEM and TOL2FEM. EQ.0: Autorecomputation off. EQ.1: Autorecomputation on.
AUTOBEM	In addition to NCYLBEM, this triggers an automatic recomputation of the BEM matrices based on an error calculation of the conductors' relative displacements. See TOL1BEM and TOL2BEM. EQ.0: Autorecomputation off. EQ.1: Autorecomputation on.

VARIABLE	DESCRIPTION
TOL1FEM	If a conducting element sees a deformation or a conductivity change that reaches an error higher than TOL1FEM, then the FEM matrices will be reassembled. If a negative value is entered, then the absolute value will refer to a load curve giving TOL1FEM function of time.
TOL2FEM	If TOL2FEM*Number-of-conducting-elements see a deformation or a conductivity change that reaches an error higher than TOL2FEM, then the FEM matrices will be recomputed. If a negative value is entered, then the absolute value will refer to a load curve giving TOL2FEM function of time.
TOL1BEM	If a conducting element sees a displacement that reaches an error higher than TOL1BEM, then the BEM matrices will be reassembled. If a negative value is entered, then the absolute value will refer to a load curve giving TOL1BEM function of time.
TOL2BEM	If TOL2BEM*Number-of-conducting-elements see a displacement that reaches an error higher than TOL2BEM, then the BEM matrices will be recomputed. If a negative value is entered, then the absolute value will refer to a load curve giving TOL2BEM function of time.

*EM_CONTROL_SWITCH

Purpose: It is possible to active a control "switch" that will shut down the solver based on a load curve information. LS-DYNA incorporates complex types of curves (See *DEFINE_CURVE_FUNCTION) that allow the setting up of complex On/Off switches, for instance, by using a nodal temperature value.

Card 1	1	2	3	4	5	6	7	8
Variable	LCID	FEMCOMP	BEMCOMP					
Type	I	I	I					
Default	0	0	0					

VARIABLE	DESCRIPTION
LCID	Load Curve ID or Define Curve Function ID. Negative values switch the solver off, positive values switch it back on.
FEMCOMP	Determines if FEM matrices are recomputed each time the EM solver is turned back on : EQ.0 : FEM matrices are recomputed EQ.1 : FEM matrices are not recomputed
BEMCOMP	Determines if BEM matrices are recomputed each time the EM solver is turned back on : EQ.0 : BEM matrices are recomputed EQ.1 : BEM matrices are not recomputed

*EM_CONTROL_SWITCH_CONTACT

Purpose: It is possible to active a control "switch" that will shut down the electromagnetic contact detection. This can be useful in order to save some calculation time in cases where the user knows when contact between conductors will occur or stop occurring.

Card 1	1	2	3	4	5	6	7	8
Variable	LCID	NCYLFEM	NCYLFEM					
Type	I	I	I					
Default	0	0	0					

VARIABLE	DESCRIPTION
LCID	Load Curve ID. Negative values switch the contact detection off, positive values switch it back on.
NCYLFEM	Determines the number of cycles before FEM matrix recomputation. If defined this will overwrite the previous NCYCLFEM as long as the contact detection is turned on.
NCYLBEM	Determines the number of cycles before BEM matrix recomputation. If defined this will overwrite the previous NCYCLBEM as long as the contact detection is turned on.

*EM_CONTROL_TIMESTEP

Purpose: Controls the EM time step and its evolution.

Card 1	1	2	3	4	5	6	7	8
Variable	TSTYPE	DTCONST	LCID	FACTOR	TSMIN	TSMAX	RLCSF	MECATS
Type	I	F	I	F	F	F	I	I
Default	none	none	none	1.0	none	none	25	0

VARIABLE	DESCRIPTION
TSTYPE	Time step type: EQ.1: Constant time step given in DTCONST EQ.2: Time step as a function of time given by a load curve specified in LCID EQ.3: Automatic time step computation, depending on the solver type. This time step is then multiplied by FACTOR EQ.4: EM time step forced to be the same as the thermal time step in problems with the EM solver coupled to the thermal solver. In other words, the thermal solver determines the EM time step.
DTCONST	Constant value for the time step for TSTYPE = 1
LCID	Load curve ID giving the time step as a function of time for TSTYPE = 2
FACTOR	Multiplicative factor applied to the time step for TSTYPE = 3
TSMIN	Minimum time step. When TSMIN is defined, the EM time step cannot drop below TSMIN. A negative value will refer to a time dependent load curve.
TSMAX	Maximum time step. When TSMAX is defined, the EM time step cannot increase beyond TSMAX. A negative value will refer to a time dependent load curve.
RLCSF	RLC Circuit time step scale factor. See Remark 2.

VARIABLE	DESCRIPTION
MECATS	Mechanical time step handling in cases where the EM solver time step becomes smaller (see Remark 3): EQ.0: Default. The EM time step will go below the solid mechanics timestep, and several EM solves will occur between two solid mechanics time steps to ensure time consistency. EQ.1: The solid mechanics time step will adapt and decrease to the EM time step value so that only one EM solve occurs between two solid mechanics solves.

Remarks:

1. **Eddy Current Solver Automatic Time Step.** For an eddy current solver, the automatic time step is based on the diffusion equation for the magnetic field:

$$\sigma \frac{\partial \vec{A}}{\partial t} + \vec{\nabla} \times \frac{1}{\mu} \vec{\nabla} \times \vec{A} + \sigma \vec{\nabla} \varphi = \vec{j}_S \ .$$

   It is computed as the minimal elemental diffusion time step over the elements. For a given element, the elemental diffusion time step is given as $dt_e = l_e^2/(2D)$, where:

   - D is the diffusion coefficient $D = 1/(\mu_0 \sigma_e)$,
   - σ_e is the element electrical conductivity,
   - μ_0 is the permeability of free space,
   - l_e is the minimal edge length of the element (minimal size of the element).

2. **Automatic Time Step with RLC Circuit.** When an automatic time step is defined and an RLC circuit is present, the EM solver will perform an additional check and calculate an approximation of the first current period based on a 0-D circuit solve. It will then limit the timestep by a factor $T_{\text{period}}/(4 \times \text{RLCSF})$. The default value of RLCSF ensures that 25 EM timesteps will be calculated for the first quarter period.

3. **MECATS.** In general, we recommend avoiding scenarios where the EM time step becomes smaller than the solid mechanics time step which are often the result of ill-defined input decks and parameters. This can, however, happen in cases where conducting elements have high deformations and an automatic EM time step is selected in which case you can choose between the two MECATS options.

*EM_DATABASE_CIRCUIT

Purpose: This keyword enables the output of EM data for every circuit defined.

Output options card

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	1	0.						

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.

Remarks:

1. The file name for this database is em_circuit_*XXX*.dat with *XXX* the circuit ID.

2. *ResistanceD* is calculated in the following way:

   a) A scalar potential difference of 1 is imposed at the circuit's boundaries SIDVIN and SIDVOUT.

   b) The system to be solved at SIDCURR is then $\nabla^2\varphi = 0$ with $\varphi_{\text{SIDVIN}} = 1$ and $\varphi_{\text{SIDVOUT}} = 0$. No diffusive effects are taken into account meaning that the current density can be written as $\mathbf{j} = \nabla\varphi$ and the total current as $I = \mathbf{j} \cdot \mathbf{n}dA$.

   c) The resistance can then be estimated using $R_D = U/I$. The calculation of this R_D resistance is solely based on the circuit's geometry and conductivity. It is therefore equivalent to the resistance as commonly defined in the circuit equations:

$$R_D = L/\sigma S$$

where L is the length of the circuit and S its surface area.

3. *ResistanceJ* is calculated by using the data provided during the EM solve : $R_J = J/I^2$ where J and I are, respectively, the joule heating and the current. Compared with *ResistanceD, ResistanceJ* is not so much a resistance calculation since it accounts for the resistive effects (when using the Eddy current solver). Rather, it corresponds to the resistance that the circuit would need in order to get the same Joule heating in the context of a circuit equation. If all EM fields are diffused or the RH solver is being used, *ResistanceJ* should be close to *ResistanceD*.

4. Only the mutual inductances between the first three circuits defined are output.

*EM_DATABASE_CIRCUIT0D

Purpose: This keyword enables the output of EM data for every circuit defined.

Output options card

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.

Remarks:

1. The file name for this database is em_circuit0D_*XXX*.dat with *XXX* the circuit ID.

2. At the start of the run, based on the initial values of the meshes resistances and inductances, the solver will calculate the results for a so-called "0D" solution which does not take into account the current's diffusion, the part's displacements or the EM material property changes. It is therefore a crude approximation. This can be useful in some cases especially in R,L,C circuits if the users wishes to have an first idea of how the source current will behave.

3. Since the calculation of this 0D circuit can take time depending on the problems size, it should only be used in cases where the output results are useful to the comprehension of the analysis.

4. This card has no influence on the results of the EM run itself.

*EM_DATABASE_ELOUT

Purpose: This keyword enables the output of EM data on elements.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	ELSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.
ELSID	Solid Elements Set ID.

Remarks:

1. The file name for this database is em_elout.dat.

*EM_DATABASE_FIELDLINE

Purpose: The EM solver uses a BEM method to calculate the EM fields between conductors. With this method, the magnetic field in the air or vacuum between conductors is therefore not explicitly computed. However, in some cases, it may be interesting to visualize some magnetic field lines for a better analysis. This keyword allows the output of field line data. It has no influence on the results of the EM solve.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	FLID	PSID	DTOUT	NPOINT				
Type	I	I	F	I				
Default	none	none	0.	100				

Remaining cards are optional.†

Card 2	1	2	3	4	5	6	7	8
Variable	INTEG	H	HMIN	HMAX	TOLABS	TOLREL		
Type	I	F	F	F	F	F		
Default	2	0.	0.	1E10	1E-3	1E-5		

Card 3	1	2	3	4	5	6	7	8
Variable	BTYPE							
Type	I							
Default	2							

VARIABLE	DESCRIPTION
FLID	Field line set ID
PSID	Point Set ID associated to the field line set (See *EM_POINT_SET). The coordinates given by the different points will be the starting points of the field lines.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM time step will be used.
NPOINT	Number of points per field line. The points are regularly spaced.
INTEG	Type of numerical integrator used to compute the field lines : EQ.1: RK4, Runge Kutta 4. See Remark 2. EQ.2: DOP853, Dormand Prince 8(5,3). See Remark 2.
H	Value of the step size. In case of an integrator with adaptive step size, it is the initial value of the step size.
HMIN	Minimal step size value. Only used in the case of an integrator with adaptive step size.
HMAX	Maximal step size value. Only used in the case of an integrator with adaptive step size.
TOLABS	Absolute tolerance of the integrator. Only used in the case of an integrator with adaptive step size.
TOLREL	Relative tolerance of the integrator. Only used in the case of an integrator with adaptive step size.
BTYPE	Method to compute the magnetic field : EQ.1: Direct method (every contribution is computed by the Biot Savart Law and summed up : very slow). EQ.2: Multipole method (approximation of the direct method using the multipole expansion). EQ.3: Multicenter method (approximation of the direct method using a weighted subset of points only in order to compute the magnetic field).

Remarks:

1. **File Names.** The file name for this database is em_fieldLine_*XX*_*YYY*.dat where *XX* is the field line ID and *YYY* is the point set ID defined in *EM_POINT_SET.

2. **Integrators.** The Runge Kutta 4 integrator is an explicit iterative method for solving ODEs. It is a fourth order method with a constant step size. The Dormand Prince 8(5,3) integrator is an explicit iterative method for solving IDEs. Particularly, this integrator is an embedded Runge Kutta integrator of order 8 with an adaptive step size. This integrator allows a step size control which is done though an error estimate at each step. The Dormand Prince 8(5,3) is a Dormand Prince 8(6) for which the 6^{th} order error estimator has been replaced by a 5^{th} order estimator with 3^{rd} order correction in order to make the integrator more robust.

*EM_DATABASE_GLOBALENERGY

Purpose: Enable the output of global EM energies.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.

Remarks:

1. **Database Name.** The file name for this database is em_globEnergy.dat.

2. **Output Data.** The output file includes the global EM energies of the mesh, the air, and the source circuit. It also includes the global kinetic energy and the global plastic work energy.

*EM_DATABASE_NODOUT

Purpose: This keyword enables the output of EM data on nodes.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	NSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.
NSID	Node Set ID.

Remarks:

1. The file name for this database is em_nodout.dat.

*EM_DATABASE_PARTDATA

Purpose: This keyword enables the output of EM data for every part defined. .

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.

Remarks:

1. The file name for this database is em_partData_*XXX*.dat with *XXX* the part ID.
2. Outputs the part EM energies of the part as well as the Lorentz force. Also outputs the part kinetic energy and the part plastic work energy.

*EM_DATABASE_POINTOUT

Purpose: This keyword enables the output of EM data on points sets.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	PSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the ICFD timestep will be used.
PSID	Point Set ID (See *EM_POINT_SET card).

Remarks:

1. The file name for this database is em_pointout.dat.

*EM_DATABASE_ROGO

Purpose: This keyword enables the output of EM data for every circuit defined. .

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	1	0.						

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the EM timestep will be used.

Remarks:

1. The file name for this database is em_rogoCoil_*XXX*.dat where *XXX* is the rogo Coil ID.

*EM_DATABASE_TIMESTEP

Purpose: This keyword enables the output of EM data regarding the EM timestep.

Output options card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.

Remarks:

1. The file name for this database is em_timestep.dat.

2. Outputs the run's EM tim estep versus the time step calculated using the EM CFL condition as criteria (autotimestep). This can be useful in cases with big deformations and/or material property changes and a fixed time step is being used in case that time step becomes to big compared to the stability time step.

*EM_EP_CELLMODEL_DEFINE_FUNCTION

Purpose: Define a user defined ionic cell model for Electro-Physiology.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	NSTATE	FSWITCH					
Type	I	I	I					
Default	none	none	none					

Card 2	1	2	3	4	5	6	7	8
Variable	DVDT	DU1DT	DU2DT	DU3DT	DU4DT	DU5DT	DU6DT	DU7DT
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

Card 3	1	2	3	4	5	6	7	8
Variable	V0	U1	U2	U3	U4	U5	U6	U7
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
MATID	Material ID defined in the *MAT section
FSWITCH	Switch for the ODE definition (see Remark 1): EQ.0: functions EQ.1: derivatives

VARIABLE	DESCRIPTION
NSTATE	Number of state variables $u_1, u_2, \dots, u_n$. The maximum value is 7 (see Cards 2 and 3).
DVDT	Function ID (see *DEFINE_FUNCTION) for evolution of V (function g in the equations in Remark 1).
DUiDT	Function ID (see *DEFINE_FUNCTION) for evolution of u_i (function f_i in the equations in Remark 1)
V0	Function ID (see *DEFINE_FUNCTION) for initial value of $V(x,y,z)$
Ui	Function ID (see *DEFINE_FUNCTION) for initial value of $u_i(x,y,z)$

Remarks:

1. This allows having a user defined cell model defined through define functions (See *DEFINE_FUNCTION). The model is composed of the transmembrane potential, V, along with n state variables $u_1, u_2, \dots u_n$. Their temporal evolution is given depending upon FSWITCH.

   a) If FSWITCH = 0:

$$
\begin{aligned}
V(t) &= g\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big) \\
u_1(t) &= f_1\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big) \\
u_2(t) &= f_2\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big) \\
&\vdots \\
u_n(t) &= f_n\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big)
\end{aligned}
$$

   b) If FSWITCH = 1:

$$
\begin{aligned}
\frac{\partial V(t)}{\partial t} &= g\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big) \\
\frac{\partial u_1(t)}{\partial t} &= f_1\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big) \\
\frac{\partial u_2(t)}{\partial t} &= f_2\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big) \\
&\vdots \\
\frac{\partial u_n(t)}{\partial t} &= f_3\big(t, dt, V(t-1), u_1(t-1), u_2(t-1), \dots, u_n(t-1)\big)
\end{aligned}
$$

2. **Benchmarks.** The electrophysiology benchmarks presented in:

[1] "Verification of computational models of cardiac electro-physiology", Pathmanathan P, Gray RA., Int J Numer Method Biomed Eng. 2014 May;30(5):525-44

can be done with this model.

*EM_EP_CELLMODEL_FENTONKARMA

Purpose: Define a Fenton-Karma ionic cell model for Electro-Physiology.

Card 1	1	2	3	4	5	6	7	8
Variable	MATID							
Type	I							
Default	none							

Card 2	1	2	3	4	5	6	7	8
Variable	TAUD	TAUR	TAUSI	TAUO	TAUVP	TAUVM	TAUWP	TAUWM
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 3	1	2	3	4	5	6	7	8
Variable	UC	UCSI	K					
Type	F	F	F					
Default	none	none	none					

Card 4	1	2	3	4	5	6	7	8
Variable	U0	V0	W0					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
MATID	Material ID defined in *MAT
TAUD	Time constant τ_d described in Equation 5
TAUR	Time constant τ_r described in Equation 6
TAUSI	Time constant τ_{si} described in Equation 7
TAU0	Time constant τ_0 described in Equation 6
TAUVP	Time constant τ_{vp} described in Equation 3
TAUVM	Time constant τ_{vm} described in Equation 3
TAUWP	Time constant τ_{wp} described in Equation 4
TAUWM	Time constant τ_{wm} described in Equation 4
UC	Threshold potential, u_c for activation of J_{fi} (the fast inward current) in Equations 3, 4, 5, and 6
UCSI	Threshold potential u_c^{si} for activation of J_{si} (the slow inward current) in Equation 7
K	Constant k in Equation 7
U0/V0/W0	Initial values of u, v, and w, respectively

Remarks:

The Fenton-Karma model is a simplified ionic model with three membrane currents that approximates well the restitution properties and spiral wave behavior of more complex ionic models of cardiac action potential (Beeler-Reuter and others). It was introduced in [1].

The total current flowing through the membrane is given by:

$$I_{\text{ion}} = -C_m \frac{\partial V}{\partial t} = -J_{\text{fi}} \tag{1}$$

where V is the transmembrane potential, C_m is the specific capacitance of the cell membrane, and J_{fi} is the fast inward current.

The model depends on three state variables, u, v, and w, and three membrane currents, J_{fi}, J_{so} (slow outward current), and J_{si} (slow inward current), through the following equations:

$$\frac{du}{dt} = -J_{\text{fi}} - J_{\text{so}} - J_{\text{si}} \tag{2}$$

$$\frac{dv}{dt} = \frac{\Theta(u_c - u)(1 - v)}{\tau_{vm}} - \frac{\Theta(u - u_c)v}{\tau_{vp}} \tag{3}$$

$$\frac{dw}{dt} = \frac{\Theta(u_c - u)(1 - w)}{\tau_{wm}} - \frac{\Theta(u - u_c)w}{\tau_{wp}} \tag{4}$$

$$J_{\text{fi}} = -\frac{\Theta(u_c - u)(1 - u)(u - u_c)}{\tau_d} \tag{5}$$

$$J_{\text{so}} = \frac{u\,\Theta(u_c - u)}{\tau_o} + \frac{u\,\Theta(u - u_c)}{\tau_r} \tag{6}$$

$$J_{\text{si}} = -\frac{w(1 + \tanh\left[k(u - u_c^{si}\right])}{2\tau_{si}} \tag{7}$$

In the above Θ is the Heaviside step function.

References:

[1] Fenton, F. & A. Karma, "Vortex dynamics in three-dimensional continuous myocardium with fiber rotation. Filament instability and fibrillations," *Chaos, Solitons, and Fractals*, Vol. 8, No. 1, pp. 661-686, (1998).

[2] https://www.ibiblio.org/e-notes/html5/fk.html

*EM_EP_CELLMODEL_FITZHUGHNAGUMO

Purpose: Define a Fitzhugh-Nagumo ionic cell model for Electro-Physiology.

Card 1	1	2	3	4	5	6	7	8
Variable	MATID							
Type	I							
Default	none							

Card 2	1	2	3	4	5	6	7	8
Variable	ALPHA	BETA	GAMMA	C	MU1	MU2		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Card 3	1	2	3	4	5	6	7	8
Variable	V	R						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
MATID	Material ID defined in *MAT
ALPHA	Excitation constant α described in Equation 1
BETA	Excitation constant β described in Equation 2
GAMMA	Excitation constant γ described in Equation 2
C	Excitation constant c described in Equation 1

VARIABLE	DESCRIPTION
MU1	Excitation constant μ_1 described in Equation 2
MU2	Excitation constant μ_2 described in Equation 2
V	Initial value of V
R	Initial value of r

Remarks:

In the Fitzhugh-Nagumo model, the excitation is defined by a cubic polynomial along with one recovery variable, r. The transmembrane current, I_{ion}, is given by:

$$I_{\text{ion}} = -C_m \frac{\partial V}{\partial t} = -cV\ (V-\alpha)(V-1) - rV \tag{1}$$

Here V is the transmembrane potential, C_m is the specific capacitance of the cell membrane, and c and α are excitation constants.

The recovery variable r evolves according to:

$$\frac{dr}{dt} = (\gamma + \frac{r\mu_1}{\mu_2 + V})(-r - cV(V-\beta-1)) \tag{2}$$

where β, γ, μ_1 and μ_2 are excitation constants.

References:

[1] Aliev, R.R. and Panfilov, A.V., "A simple two-variable model of cardiac excitation," *Chaos, Solitons, and Fractals*, Vol 7, No 3, pp 293-301, (1996).

[2] Pullan, A.J., Cheng, L.K., and Buist, M.L., Mathematically Modelling the Electrical Activity of the Heart, World Scientific Publishing Co. Pte. Ltd., Singapore, pp 132-133, (2005).

[3] Baillargeon, B. et al., "The Living Heart Project: A robust and integrative simulator for human heart function," *European Journal of Mechanics - A/Solids*. Vol 48, pp 38-47, (2014).

*EM_EP_CELLMODEL_TENTUSSCHER

Purpose: Define a ten Tusscher ionic cell model for Electro-Physiology.

Card Summary:

Card 1. This card is required.

MID							

Card 2. This card is required.

R	T	F	CM	VC	VSR	VSS	PKNA

Card 3. This card is required.

KO	NAO	CAO					

Card 4. This card is required.

GK1	GKR	GKS	GNA	GBNA	GCAL	GBCA	GTO

Card 5. This card is required.

GPCA	GPK						

Card 6. This card is required.

PNAK	KMK	KMNA	KNACA	KSAT	ALPHA	GAMMA	KMCA

Card 7. This card is required.

KMNAI	KPCA						

Card 8. This card is required.

K1	K2	K3	K4	EC	MAXSR	MINSR	

Card 9. This card is required.

VREL	VLEAK	VXFER	VMAXUP	KUP			

Card 10. This card is required.

BUFC	KBUFC	BUFSR	KBUFSF	BUFSS	KBUFSS		

Card 11. This card is required.

V	KI	NAI	CAI	CASS	CASR	RPRI	

Card 12. This card is required.

XR1	XR2	XS	M	H	J	D	F

Card 13. This card is required.

F2	FCASS	S	R				

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	MID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MID	Material ID defined in *MAT section

Card 2	1	2	3	4	5	6	7	8
Variable	R	T	F	CM	VC	VSR	VSS	PKNA
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
R	Gas constant ($J \times K^{-1} \times Mol^{-1}$)
T	Temperature (K)

VARIABLE	DESCRIPTION
F	Faraday constant ($C \times mmol^{-1}$)
CM	Cell capacitance for unit surface area ($\mu F \times Cm^{-2}$)
VC	Cytoplasmic volume (μm^3)
VSR	Sarcoplasmic reticulum volume (μm^3)
VSS	Subspace volume (μm^3)
PKNA	Relative I_{Ks} permeability to Na^+

Card 3	1	2	3	4	5	6	7	8
Variable	KO	NAO	CAO					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
KO	Extracellular K^+ concentration (mM)
NAO	Extracellular Na^+ concentration (mM)
CAO	Extracellular Ca^{2+} concentration (mM)

Card 4	1	2	3	4	5	6	7	8
Variable	GK1	GKR	GKS	GNA	GBNA	GCAL	GBCA	GTO
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 5	1	2	3	4	5	6	7	8
Variable	GPCA	GPK						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
GK1, GKR, GKS, GNA, GBNA, GCAL, GBCA, GTO, GPCA, GPK	Maximal I_{K1}, I_{Kr}, I_{Ks}, I_{Na}, I_{bNa}, I_{CaL}, I_{bCa}, I_{to}, I_{pCa}, and I_{pK} conductance, respectively (units: nS × pF^{-1})

Card 6	1	2	3	4	5	6	7	8
Variable	PNAK	KMK	KMNA	KNACA	KSAT	ALPHA	GAMMA	KMCA
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 7	1	2	3	4	5	6	7	8
Variable	KMNAI	KPCA						
Type	F	F						
Default	none	none						

VARIABLE	DESCRIPTION
PNAK	P_{NaK}, parameter for calculating the Na^+/K^+ pump current (units: pA × pF^{-1}). See Reference [1].

VARIABLE	DESCRIPTION
KMK, KMNA	K_{mK} and K_{mNa}, parameters for calculating the Na^+/K^+ pump current (units: millimolar). See Reference [1].
KNACA, KSAT, ALPHA, GAMMA, KMNAI	k_{NaCa}, k_{sat}, α, γ, and K_{mNai}, parameters for calculating the Na^+/Ca^{2+} exchanger current (units: millimolar). See Reference [1].
KPCA	K_{pCa}, parameter for calculating Ca^{2+} pump current (units: millimolar). See Reference [1].

Card 8	1	2	3	4	5	6	7	8
Variable	K1	K2	K3	K4	EC	MAXSR	MINSR	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	
Ref	2	2	2	2	2	2	2	

VARIABLE	DESCRIPTION
K1	R to O and RI to I I_{rel} transition rate ($mM^{-2} \times ms^{-1}$)
K2	O to I and R to RI I_{rel} transition rate ($mM^{-1} \times ms^{-1}$)
K3	O to R and I to RI I_{rel} transition rate (ms^{-1})
K4	I to O and RI to I I_{rel} transition rate (ms^{-1})
EC	Ca_{SR} half-saturation constant of k_{casr} (mM)
MAXSR/MINSR	Maximum and minimum values of k_{casr}

Card 9	1	2	3	4	5	6	7	8
Variable	VREL	VLEAK	VXFER	VMAXUP	KUP			
Type	F	F	F	F	F			
Default	none	none	none	none	none			

VARIABLE	DESCRIPTION
VREL, VLEAK, VXFER, VMAXUP	Maximal I_{rel}, I_{leak}, I_{xfer}, and I_{up} conductance (mM × ms^{-1}), respectively. See Reference [2].
KUP	Half-saturation constant of I_{up} (mM). See Reference [2].

Card 10	1	2	3	4	5	6	7	8
Variable	BUFC	KBUFC	BUFSR	KBUFSF	BUFSS	KBUFSS		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		
Ref	2	2	2	2	2	2		

VARIABLE	DESCRIPTION
BUFC	Total cytoplasmic buffer concentration (mM)
KBUFC	Ca_i half-saturation constant for cytoplasmic buffer (mM)
BUFSR	Total sarcoplasmic buffer concentration (mM)
KBUFSR	Ca_{SR} half-saturation constant for sarcoplasmic buffer (mM)
BUFSS	Total subspace buffer concentration (mM)
KBUFSS	Ca_{SS} half-saturation constant for subspace buffer (mM)

Card 11	1	2	3	4	5	6	7	8
Variable	V	KI	NAI	CAI	CASS	CASR	RPRI	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	
Ref	2	1	1	2	2	2	2	

VARIABLE	DESCRIPTION
V	Initial value of transmembrane potential (mV)
KI	Initial value of K_i, used in potassium dynamics (mM)
NAI	Initial value of Na_i, used in sodium dynamics (mM)
CAI	Initial value of Ca_i. used in calcium dynamics (mM)
CASS	Initial value of Ca_{SS}, used in calcium dynamics (mM)
CASR	Initial value of Ca_{SR}, used in calcium dynamics (mM)
RPRI	Initial value of R', used in calcium dynamics

Card 12	1	2	3	4	5	6	7	8
Variable	XR1	XR2	XS	M	H	J	D	F
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none
Ref	1	1	1	1	1	1	2	2

VARIABLE	DESCRIPTION
XR1	Initial value of x_{r1}, used to compute the rapid time dependent K^+ current

VARIABLE	DESCRIPTION
XR2	Initial value of x_{r2}, used to compute the rapid time dependent K^+ current
XS	Initial value of x_s, used to compute slow time dependent K^+ current
M	Initial value of m, used to compute the fast Na^+ current
H	Initial value of h, used to compute the fast Na^+ current
J	Initial value of j, used to compute the fast Na^+ current
D	Initial value of d, used to compute the L-type Ca^{2+} current
F	Initial value of f, used to compute the L-type Ca^{2+} current

Card 13	1	2	3	4	5	6	7	8
Variable	F2	FCASS	S	R				
Type	F	F	F	F				
Default	none	none	none	none				
Ref	2	2	1	1				

VARIABLE	DESCRIPTION
F2	Initial value of f_2, used to compute the L-type Ca^{2+} current
FCASS	Initial value of f_{cass}, used to compute the L-type Ca^{2+} current
S	Initial value of s, used to compute the transient outward current
R	Initial value of r, used to compute the transient outward current

Remarks:

This is a model of the action potential of human ventricular cells that, while including a high level of electrophysiological detail, is computationally cost-effective enough to be applied in large-scale spatial simulations for the study of reentrant arrhythmias. Please see the references for details. This model is based on [2].

References:

[1] "A model for human ventricular tissue", K.H.W.J. ten Tusscher et Al., Am J Physiol Heart Circ Physiol 286: H1573-H1589, 2004

[2] "Alternans and spiral breakup in human ventricular tissue model", K.H.W.J. ten Tusscher and A.V. Panfilov, Am J Physiol Heart Circ Physiol 291: H1088-H1100, 2006

*EM_EP_CELLMODEL_TOMEK

Purpose: Define a ToR-ORd model for cardiac electrophysiology [1].

> **WARNING:** As of version R15, this keyword is deprecated. Instead, use *EM_EP_CELLMODEL_TOR_ORD which is an updated version of this keyword.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	PHIEND	PHIMYO					
Type	I	F	F					
Default	none	0.0	0.0					

VARIABLE	DESCRIPTION
MID	Material ID: refers to MID in the *PART card.
PHIEND	Value between 0.0 and 1.0 giving the ratio of the cardiac tissue to be considered endocardial in the ToR-Ord cell model.
PHIMYO	Value between 0.0 and 1.0 giving the ratio of the cardiac tissue to be considered myocardial in the ToR-Ord cell model.

References:

[1] Tomek J., Bueno-Orovio A., Passini E., Zhou X., Minchole A., Britton O., Bartolucci C., Severi S., Shrier A., Virag L., Varro A., and Rodriguez B., “Development, calibration, and validation of a novel human ventricular myocyte model in health, disease, and drug block,” Elife (2019).

*EM_EP_CELLMODEL_TOR_ORD

Purpose: Define a ToR-ORd model for cardiac electrophysiology [1].

NOTE: This keyword is an updated version of *EM_EP_CELLMODEL_-TOMEK. For versions R15 and later, *EM_EP_CELLMODEL_-TOR_ORD is the preferred keyword, and *EM_EP_CELLMOD-EL_TOMEK is deprecated.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	LDID	PHIEND	PHIMYO				
Type	I	I	F	F				
Default	none	none	0.0	0.0				

VARIABLE	DESCRIPTION
MID	Material ID. It refers to MID in the *PART card.
LDID	ID of the *EM_EP_LAPLACE_DIRICHLET solution to be used for transmural depth definition
PHIEND	Value between 0.0 and 1.0 giving the ratio of the cardiac tissue to be considered endocardial in the ToR-Ord cell model.
PHIMYO	Value between 0.0 and 1.0 giving the ratio of the cardiac tissue to be considered myocardial version in the ToR-Ord cell model.

References:

[1] Tomek J., Bueno-Orovio A., Passini E., Zhou X., Minchole A., Britton O., Bartolucci C., Severi S., Shrier A., Virag L., Varro A., and Rodriguez B., “Development, calibration, and validation of a novel human ventricular myocyte model in health, disease, and drug block,” Elife (2019).

*EM_EP_CELLMODEL_USERMAT

Purpose: Specify a user material for an ionic cell model to be used in electrophysiology simulations.

Card 1	1	2	3	4	5	6	7	8
Variable	MID							
Type	I							

VARIABLE	DESCRIPTION
MID	Material ID. A unique number must be specified (see *PART).

*EM_EP_CREATEFIBERORIENTATION

Purpose: Define fiber orientation by solving a Laplace-Dirichlet system defined by *EM_EP_LAPLACE_DIRICHLET. This feature is based on [1].

Card 1	1	2	3	4	5	6	7	8
Variable	PSID	LDID1	LDID2	ALPHA	BETA	IEXPORT	IPRERUN	
Type	I	I	I	F	F	I	I	
Default	none	none	none	0.0	0.0	0	0	

VARIABLE	DESCRIPTION
PSID	Part set ID of the part set on which the system is solved
LDID1	ID of the Laplace system that is solved in the transmural direction
LDID2	ID of the Laplace system that is solved in the apicobasal direction
ALPHA	Helical angle with respect to the counterclockwise circumferential direction in the heart when looking from the base towards the apex. LT.0: |ALPHA| is the ID for the *DEFINE_FUNCTION giving the helical angle. See Remark 1 for available arguments.
BETA	Angle with respect to the outward transmural axis of the heart. LT.0: |BETA| is the ID for the *DEFINE_FUNCTION giving the angle. See Remark 1 for available arguments.
IEXPORT	Selects whether result files (ELEMENT_SOLID_ORTHO.k and vtk files) are exported: EQ.0: Not exported EQ.1: Exported
IPRERUN	Select whether the run is stopped after creating fibers: EQ.0: Do not stop after fiber creation EQ.1: Stop after fiber creation

Remarks:

1. ***DEFINE_FUNCTION Arguments.** The arguments for the function defined with *DEFINE_FUNCTION may include the following: f(x_ele, y_ele, z_ele, phi_len, phi_thi). Here phi_len and phi_thi are the potentials corresponding to the potentials solved in LDID1 and LDID2, respectively.

References:

[1] Bayer, J.D., Blake, R. C., Plank, G., and Trayanova, N. A., "A novel rule-based algorithm for assigning myocardial fiber orientation to computational heart models," Annals of biomedical engineering, 40(10), 2243-2254 (2012).

*EM_EP_ECG

Purpose: Compute pseudo-ECGs on a set of virtual points. LS-DYNA exports a file named em_ECG_{ECGID}.dat.

Card 1	1	2	3	4	5	6	7	8
Variable	ECGID	PSID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
ECGID	ID of the ECG computation
PSID	Point set ID containing the list of virtual points on which the pseudo-ECGs are computed

*EM_EP_ISOCH

Purpose: Compute isochrones of activation/repolarization of a cardiac electrophysiology wave of one or several beats.

Card 1	1	2	3	4	5	6	7	8
Variable	idISOCH	IDEPOL	DPLTHR	IREPOL	RPLTHR			
Type	I	I	F	I	F			
Default	none	0	none	0	none			

VARIABLE	DESCRIPTION
idISOCH	Id of the isochrone
IDEPOL	Flag to activate the computation of depolarization: EQ.0: Off EQ.1: On
DPLTHR	Amplitude threshold used for measuring depolarization
IREPOL	Flag to activate the computation of repolarization times: EQ.0: Off EQ.1: On
RPLTHR	Amplitude threshold used for measuring repolarization

*EM_EP_LAPLACE_DIRICHLET

Purpose: Define the boundary conditions of the Laplace-Dirichlet system to be solved to define fiber orientation, based on [1]. The system is solved by using *EM_EP_CREATE-FIBERORIENTATION. This keyword was formerly called *EM_EP_FIBERINITIAL in versions R14 and earlier.

Include as many of this card as needed. The next keyword ("*") card terminates this input.

Card 1	1	2	3	4	5	6	7	8
Variable	LDID	PID	STYPE	SID1	SID0			
Type	I	I	I	I	I			
Default	none	none	none	none	none			

VARIABLE	DESCRIPTION
LDID	ID of the Laplace system to solve (define a new ID with each new card)
PID	Part ID on which the system is solved
STYPE	Set type for the boundary condition: EQ.1: Segment set EQ.2: Node set
SID1	Set on which a potential of value 1 is prescribed
SID0	Set on which a potential of value 0 is prescribed

References:

[1] Bayer, J.D., Blake, R. C., Plank, G., and Trayanova, N. A., "A novel rule-based algorithm for assigning myocardial fiber orientation to computational heart models," Annals of biomedical engineering, 40(10), 2243-2254 (2012).

*EM_EP_PURKINJE_NETWORK

Purpose: Define a Purkinje network that consists of conductive beams and lies on a given surface, based on Costabal et al [2016].

Card 1	1	2	3	4	5	6	7	8
Variable	PURKID	BUILDNET	SSID	MID	POINTSTX	POINTSTY	POINTSTZ	EDGELEN
Type	I	I	I	I	F	F	F	F

Card 2	1	2	3	4	5	6	7	8
Variable	NGEN	NBRINIT	NSPLIT	INODEID	IEDGEID			
Type	I	I	I	I	I			

VARIABLE	DESCRIPTION
PURKID	ID for the Purkinje network
BUILDNET	Flag to create Purkinje network: EQ.0: Purkinje network not created. EQ.1: New Purkinje network created.
SSID	Segment set on which the Purkinje network is lying
MID	Material ID defined in the *MAT section.
POINTSTX	*X* coordinate of the tree origin
POINTSTY	*Y* coordinate of the tree origin
POINTSTZ	*Z* coordinate of the tree origin
EDGELEN	Edge length
NGEN	Number of generations of branches
NBRINIT	Number of branches attached to the tree origin

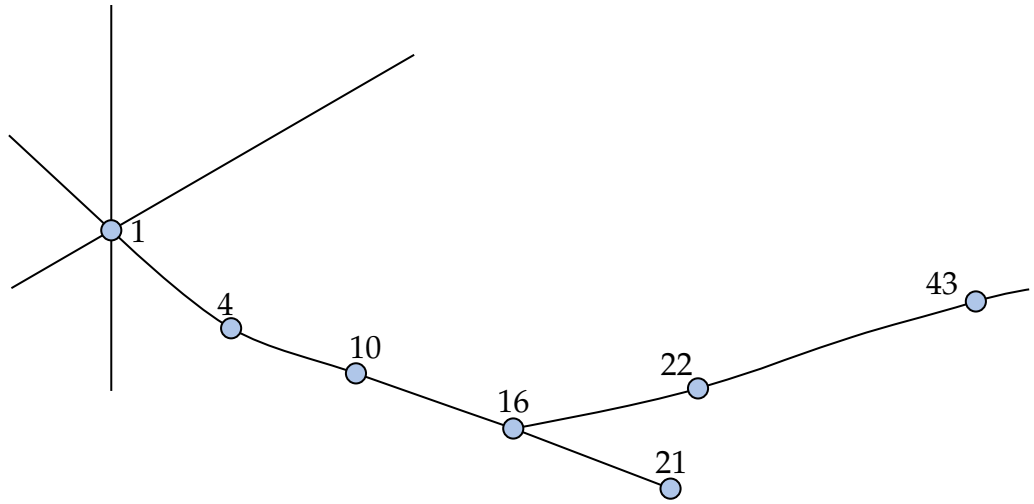


Figure 6-3. Example of part of a network. For this network, NSPLIT = 4.

VARIABLE	DESCRIPTION
NSPLIT	Number of nodes between two consecutive branchings as shown in Figure 6-3
INODEID	Initial node ID
IEDGEID	Initial edge ID. These edges are internally generated by LS-DYNA.

References:

Costabal, F.S., D. E. Hurtado, and E. Kuhl, "Generating Purkinje networks in the human heart," Journal of Biomechanics, vol. 49, issue 12, pp. 2455–2465, (2016).

*EM_EP_TENTUSSCHER_STIMULUS

Purpose: Define a stimulation pattern of a Tentusscher cell model (requires the use of *EM_EP_CELLMODEL_TENTUSSCHER).

Card 1	1	2	3	4	5	6	7	8
Variable	STIMID	SETTYPE	SETID					
Type	I	I	I					

Card 2	1	2	3	4	5	6	7	8
Variable	STIMSTRT	STIMT	STIMDUR	STIMAMP				
Type	F	F	F	F				

VARIABLE	DESCRIPTION
STIMID	ID of the stimulation
SETTYPE	Set type: EQ.1: Segment set EQ.2: Node set
SETID	Node set or segment set ID to be stimulated
STIMSTRT	Starting time of the stimulation
STIMT	Stimulation period
STIMDUR	Stimulation duration
STIMAMP	Stimulation amplitude (picoA/picoF)

*EM_EOS_BURGESS

Purpose: Define the parameters for a Burgess model giving the electrical conductivity as as a function of the temperature and the density, see:

T.J. Burgess, "Electrical resistivity model of metals", 4th International Conference on Megagauss Magnetic-Field Generation and Related Topics, Santa Fe, NM, USA, 1986

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	V0	GAMMA	THETA	LF	C1	C2	C3
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	C4	K	EXPON	LGTUNIT	TIMUNIT	TEMUNI	ADJUST	
Type	F	F	I	F	F	I	I	
Default	none	none	none	none	none	none	none	

In the following, UUS stands for User Units System and BUS for Burgess Units

VARIABLE	DESCRIPTION
EOSID	ID of the EM_EOS (specified by an *EM_MAT card)
V0	Reference specific volume V_0 (UUS).
GAMMA0	Reference Gruneisen value γ_0.(no units).
THETA	Reference melting temperature $\theta_{m,0}$ in eV (BUS).
LF	Latent heat of fusion L_F in kJoule/mol (BUS).
C1	C1 constant (BUS)
C2	C2 constant (no units)

VARIABLE	DESCRIPTION
C3	C3 constant (no units)
C4	C4 constant (no units)
K	Parameter k (no units).
EXPON	Exponent in equations (2) (see remarks)
LGTUNIT	Length units for UUS (relative to meter, i.e. = 1.e-3 if UUS in mm).
TIMUNIT	Time units for UUS (relative to seconds).
TEMUNIT	Temperature units EQ.1: temperature in Celsius EQ.2: temperature in Kelvins
ADJUST	Conductivity modification EQ.0: (default) The conductivity is given by the Burgess formula. EQ.1: The conductivity is adjusted so that it is equal to the conductivity defined in *EM_MAT card σ_{mat} at room temperature: $$\sigma(\theta) = \sigma_{\text{Burgess}}(\theta)\frac{\sigma_{\text{mat}}}{\sigma_{\text{Burgess}}(\theta_{\text{room}})}$$

Remarks:

1. The Burgess model gives the electrical resistivity vs temperature and density for the solid phase, liquid phase and vapor phase. At this time, only the solid and liquid phases are implemented. To check which elements are in the solid and in the liquid phase, a melting temperature is first computed by:

$$\theta_m = \theta_{m,0}\left(\frac{V}{V_0}\right)^{-\frac{1}{3}} e^{(2\gamma_0-1)(1-\frac{V}{V_0})}$$

   a) If T < θ_m: solid phase model applies.

   The solid phase electrical resistivity corresponds to the Meadon model:

$$\eta_S = (C_1 + C_2\theta^{C_3})f_c\left(\frac{V}{V_0}\right), \quad (1)$$

where θ is the temperature, V is the specific volume, and V_0 is the reference specific volume (zero pressure, solid phase). In (1), the volume dependence is given by:

$$f_c\left(\frac{V}{V_0}\right) = \begin{cases} \left(\frac{V}{V_0}\right)^{2\gamma-1} & \text{EXPON.EQ.} -1 \quad \text{(most materials)} \\ \left(\frac{V}{V_0}\right)^{2\gamma+1} & \text{EXPON.EQ.} +1 \quad \text{(tungsten)} \\ \left(\frac{V}{V_0}\right)^{2\gamma} & \text{EXPON.EQ.} 0 \quad \text{(stainless steel)} \end{cases} \tag{2}$$

with

$$\gamma = \gamma_0 - \left(\gamma_0 - \frac{1}{2}\right)\left(1 - \frac{V}{V_0}\right) \tag{3}$$

b) If $T > \theta_m$: liquid phase model:

$$\eta_L = (\eta_L)_{\theta_m}\left(\frac{\theta}{\theta_m}\right)^{C_4} \tag{4}$$

with

$$(\eta_L)_{\theta_m} = \Delta\eta(\eta_S)_{\theta_m}$$

where

$$\Delta\eta = \begin{cases} ke^{0.69L_F/\theta_m} & k > 0 \\ 1 + 0.0772(2 - \theta_m) & k = -1 \quad \text{(tungsten)} \\ 1 + 0.106(0.846 - \theta_m) & k = -2 \quad \text{(stainless steel SS-304)} \end{cases} \tag{5}$$

The following table reports some sets of parameters given by Burgess in his paper:

Parameter	Cu	Ag	Au	W	Al(2024)	SS(304)
V_0(cm³/gm)	0.112	0.0953	0.0518	0.0518	0.370	0.1265
γ_0	2.00	2.55	3.29	1.55	2.13	2.00
$\theta_{m,0}$ (BUS)	0.117	0.106	0.115	0.315	0.0804	0.156
L_F (BUS)	0.130	0.113	0.127	0.337	0.107	0.153
C_1 (BUS)	-4.12e-5	-3.37e-5	-4.95e-5	-9.73e-5	-5.35e-5	0

Parameter	Cu	Ag	Au	W	Al(2024)	SS(304)
C_2	0.113	0.131	0.170	0.465	0.233	0.330
C_3	1.145	1.191	1.178	1.226	1.210	0.4133
EXPON	-1	-1	-1	+1	-1	0
C_4	0.700	0.672	0.673	0.670	0.638	0.089
k	0.964	0.910	1.08	-1.	0.878	-2.

*EM_EOS_MEADON

Purpose: Define the parameters for a Meadon model, giving the electrical conductivity as a function of the temperature and the density; see:

T.J. Burgess, "Electrical resistivity model of metals", 4th International Conference on Megagauss Magnetic-Field Generation and Related Topics, Santa Fe, NM, USA, 1986

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	C1	C2	C3	TEMUNI	V0	GAMMA	EXPON
Type	I	F	F	F	I	F	F	I
Default	none	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	LGTUNIT	TIMUNIT	ADJUST					
Type	F	F	I					
Default	none	none	none					

In the following, UUS stands for User Units System and BUS for Burgess Units.

VARIABLE	DESCRIPTION
EOSID	ID of the EM_EOS
C1	C1 constant (BUS)
C2	C2 constant (no units)
C3	C3 constant (no units)
TEMUNIT	Temperature units EQ.1: temperature in Celsius EQ.2: temperature in Kelvins

VARIABLE	DESCRIPTION
V0	Reference specific volume V0 (UUS).
GAMMA0	Reference Gruneisen value γ_0.(no units).
EXPON	Exponent in equations (7)
LGTUNIT	Length units for UUS (relative to meter, i.e. = 1.e-3 if UUS in mm).
TIMUNIT	Time units for UUS (relative to seconds).
ADJUST:	EQ.0: (default) the conductivity is given by the Burgess formula. EQ.1: The conductivity is adjusted so that it is equal to the conductivity defined in the *EM_MAT card σ_{mat}at room temperature: $$\sigma(\theta) = \sigma_{\text{Burgess}}(\theta)\frac{\sigma_{\text{mat}}}{\sigma_{\text{Burgess}}(\theta_{\text{room}})}$$

Remarks:

1. The Meadon model is a simplified Burgess model with the solid phase equations only.

   The electrical resistivity is given by:

   $$\eta_S = \left(C_1 + C_2\theta^{C_3}\right) f_c\left(\frac{V}{V_0}\right) \tag{6}$$

   where θ is the temperature, V is the specific volume, and V_0 is the reference specific volume (zero pressure, solid phase).

   In (6), the volume dependence is given by:

   $$f_c\left(\frac{V}{V_0}\right) = \begin{cases} \left(\frac{V}{V_0}\right)^{2\gamma-1} & \text{EXPON.EQ.}-1 \quad \text{(most materials)} \\ \left(\frac{V}{V_0}\right)^{2\gamma+1} & \text{EXPON.EQ.}+1 \quad \text{(tungsten)} \\ \left(\frac{V}{V_0}\right)^{2\gamma} & \text{EXPON.EQ.0} \quad \text{(stainless steel)} \\ 1 & \text{VO.EQ. 0} \quad \text{(default value for } V_0 \text{ is zero)} \end{cases} \tag{7}$$

   (In this last case, only EOSID, C1, C2, C3, TEMUNIT, TIMUNIT and LGTUNIT need to be defined)

   with,

$$\gamma = \gamma_0 - \left(\gamma_0 - \frac{1}{2}\right)\left(1 - \frac{V}{V_0}\right) \quad (8)$$

The following table reports some sets of parameters given by Burgess in his paper:

Parameter	Cu	Ag	Au	W	Al(2024)	SS(304)
V_0(cm³/gm)	0.112	0.0953	0.0518	0.0518	0.370	0.1265
γ_0	2.00	2.55	3.29	1.55	2.13	2.00
C_1 (BUS)	-4.12e-5	-3.37e-5	-4.95e-5	-9.73e-5	-5.35e-5	0
C_2	0.113	0.131	0.170	0.465	0.233	0.330
C_3	1.145	1.191	1.178	1.226	1.210	0.4133
EXPON	-1	-1	-1	+1	-1	0

*EM_EOS_PERMEABILITY

Purpose: Define the parameters for the behavior of a material's permeability

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	EOSTYPE	LCID					
Type	I	I	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
EOSID	ID of the EM_EOS
EOSTYPE	Define the type of EOS: EQ.1: Permeability defined by a B function of H curve ($B = \mu H$) EQ.2: Permeability defined by a H function of B curve ($H = B/\mu$)
LCID	Load curve ID

*EM_EOS_TABULATED1

Purpose: Define the electrical conductivity or permeability depending on the material referencing this EOS as a function of temperature by using a load curve.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	LCID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
EOSID	ID of the EM EOS
LCID	Load curve ID. See Remark 1.

Remarks:

1. **Suggestions for the Load Curve.** The load curve describes the electrical conductivity or permeability (ordinate) as a function of the temperature (abscissa). You need to make sure the temperature and the electrical conductivity / permeability given by the load curve are in the correct units. Also, we advise giving some bounds to the load curve (conductivities / permeabilities at very low and very high temperatures) to avoid bad extrapolations of the conductivity if the temperature gets out of the load curve bounds.

*EM_EOS_TABULATED2

Purpose: Define the electrical conductivity or permeability (depending on the material model referencing this EOSID) as a function of time by using a load curve.

Card 1	1	2	3	4	5	6	7	8
Variable	EOSID	LCID	IFLAG					
Type	I	I	I					
Default	none	none	0					

VARIABLE	DESCRIPTION
EOSID	ID of the EM EOS
LCID	Load curve ID (see Remark 1), function ID (see *DEFINE_FUNCTION), table ID or 2D table ID. For the arguments for the *DEFINE_FUNCTION, see Remark 2.
IFLAG	If LCID is a table ID or 2D table ID, conductivity / permeability is a function of temperature and material density. This flag dictates how LS-DYNA interprets the table. In other words, it specifies which property (temperature or material density) is the value for the table and which is the ordinate for load curves in the table: EQ.0: Temperature (value) indexes each conductivity/permeability (ordinate) versus material density (abscissa) load curve. EQ.1: Material density indexes each conductivity/permeability (ordinate) versus temperature (abscissa) load curve.

Remarks:

1. **Suggestions for the Load Curve.** The load curve describes the electrical conductivity / permeability (ordinate) as a function of time (abscissa). You need to make sure the time and the electrical conductivity / permeability given by the load curve are in the correct units. Also, we advise giving some bounds to the load curve (conductivities / permeabilities at $t = 0$ and after a long time) to avoid bad extrapolations of the conductivity / permeability if the run time gets out of the load curve bounds.

2. ***DEFINE_FUNCTION Arguments.** LCID can also refer to a *DEFINE_FUNCTION ID. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(vx, vy, vz, temp, pres, vol, mass, Ex, Ey, Ex, Bx, By, Bz, Fx, Fy, Fz, JHrate, time, x, y, z). (Fx, Fy, Fz) refers to the Lorentz force vector.

*EM_EXTERNAL_FIELD

Purpose: Define the components of a time dependent exterior field uniform in space applied on the conducting parts.

Card 1	1	2	3	4	5	6	7	8
Variable	FIELDID	FTYPE	FDEF	LCIDX	LCIDY	LCIDZ		
Type	I	I	F	I	I	I		
Default	0	0	0	0	0	0		

VARIABLE	DESCRIPTION
FIELDID	External Field ID
FTYPE	Field type: EQ.1: magnetic field EQ.2: electric field (not available yet) EQ.3: charge density (resistive heating solver only)
FDEF	Field defined by: EQ.1: load curves EQ.2: define function (FTYPE = 3 only). If a define function is used, the following parameters are accepted: x, y, z, time, emdt, pot, curr, sigma.
LCID[X,Y,Z]	Load curve ID defining the (X, Y, Z) component of the field function of time for FTYPE = 1. For FTYPE = 3, only LCIDY is used and should be a simple a load curve or define function ID.

Remarks:

1. **Electrostatic Problems.** FTYPE = 3 is mostly used in electrostatic problem configurations. The material's conductivity then represents the permittivity.

*EM_ISOPOTENTIAL

Purpose: Define an isopotential. In other words, constrain nodes so that they have the same scalar potential value. This keyword card can only be used with the type 3 EM solver.

Card 1	1	2	3	4	5	6	7	8
Variable	ISOID	SETTYPE	SETID	RDLTYPE				
Type	I	I	I	I				
Default	none	none	none	0				

VARIABLE	DESCRIPTION
ISOID	ID of the isopotential
SETTYPE	Set type: EQ.1: Segment Set EQ.2: Node Set EQ.3: Fluid surface part. See *ICFD_PART.
SETID	Set ID
RDLTYPE	Used for the battery application (see *EM_RANDLES_BATMAC or *EM_RANDLES_TSHELL). Selects which layers of the underlying battery cell are associated with the isopotential: EQ.0: Default. No specific treatment. EQ.1: Current Collector Positive EQ.2: Positive Electrode EQ.3: Separator EQ.4: Negative Electrode EQ.5: Current Collector Negative The function of a layer is defined in *EM_MAT_001.

*EM_ISOPOTENTIAL_CONNECT

Purpose: Define a connection between two isopotentials or between an isopotential and the ground.

Card 1	1	2	3	4	5	6	7	8
Variable	CONID	CONTYPE	ISOID1	ISOID2	VAL	LCID/RDLID	PSID	
Type	I	I	I	I	F	I	I	
Default	none	none	none	none	none	none	none	

R,L,C circuit parameters. Only to be defined if CONTYPE = 6.

Card 2	1	2	3	4	5	6	7	8
Variable	L	C	V0					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
CONID	Connection ID
CONTYPE	Connection type : EQ.1: Short Circuit. EQ.2: Resistance. EQ.3: Voltage Source. EQ.4: Current Source. EQ.5: Meshless Randles circuit (used to represent a cell by one lumped Randles circuit) EQ.6: R, L, C circuit
ISOID1	ID of the first isopotential to be connected

VARIABLE	DESCRIPTION
ISOID2	Optional ID of the second isopotential to be connected
VAL	Value of the resistance, voltage or current depending on CONTYPE Ignored if LCID defined.
LCID /RDLID	Load curve ID defining the value of the resistance, voltage or current function of time. If a negative value is entered, a *DEFINE_FUNCTION will be expected. The following parameters are allowed: *(time, emdt, curr1, curr2, pot1, pot2, rmesh). Pot1* and *pot2* or *curr1*, *curr2* are the potential and current at the previous timestep and two previous timesteps ago. *Rmesh* is the mesh resistance calculated by the solver at this isopot. ID of the Randles circuit defined by *EM_RANDLES_MESHLESS if CONTYPE = 5.
PSID	Used for the application: meshless Randles circuit (CONTYPE = 5) if the variable R0TOTH of *EM_RANDLES_MESHLESS is equal to 1. Part Set ID where the joule heating corresponding to the resistance r0 in *EM_RANDLES_MESHLESS is added, averaged over the part set.
L/C/V0	Circuit inductance, capacity and initial voltage. Resistance is given by VAL.

*EM_ISOPOTENTIAL_ROGO

Purpose: measures the total current flowing through a given section of the conductor and outputs it in an ASCII file called em_rogoCoil.dat

Card 1	1	2	3	4	5	6	7	8
Variable	ISOID	SETTYPE	SETID					
Type	I	I	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
ISOID	ID of the Rogo coil.
SETTYPE	Set type: EQ.1: Segment Set.
SETID	Set ID

*EM_MAT_001

Purpose: Define the electromagnetic material type and properties for a material whose permeability equals the free space permeability.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MTYPE	SIGMA	EOSID		EPLAMDA	DEATHT	RDLTYPE
Type	I	I	F	I		F	F	I
Default	none	none	none	none		optional	10^{28}	none

VARIABLE	DESCRIPTION
MID	Material ID. A unique number must be used (see *PART).
MTYPE	Defines the electromagnetism type of the material: EQ.0: Air or vacuum EQ.1: Insulator material. These materials have the same electromagnetism behavior as MTYPE = 0. EQ.2: Conductor carrying a source. In these conductors, the eddy current problem is solved, which gives the actual current density. Typically, this material would correspond to the coil. EQ.3: Fluid conductor. In this case, MID refers to the ID given in *ICFD_PART. See Remark 1. EQ.4: Conductor not connected to any current or voltage source, where the Eddy current problem is solved. Typically, this material would correspond to the workpiece.
SIGMA	Initial electrical conductivity of the material
EOSID	Optional ID of the EOS to be used for the electrical conductivity (see *EM_EOS cards).
EPLAMDA	Optional. When defined, this field activates the computation of extracellular potentials in the purkinje network with the anisotropy ratio given by EPLAMDA.

VARIABLE	DESCRIPTION
DEATHT	Death time for the material. After DEATHT, the material will no longer be considered a conductor and removed from the EM solve. If a negative value is entered, a *DEFINE_FUNCTION will be expected. The following parameters are allowed: (vx, vy, vz, temp, vol, mass, Ex, Ey, Ez, Bx, By, Bz, Fx, Fy, Fz, JHrate, time). Fx, Fy, Fz refers to the Lorentz force vector. A negative value returned by the *DEFINE_FUNCTION corresponds to a "dead" or inactive element. Once an element has been removed from the EM solve, it cannot return.
RDLTYPE	Used for composite Tshell batteries modeled with *EM_RANDLES_TSHELL. RDLTYPE specifies the function of the layer associated to MID: EQ.0: Default. Conductor which is not part of a battery cell. EQ.1: Current Collector Positive EQ.2: Positive Electrode EQ.3: Separator EQ.4: Negative Electrode EQ.5: Current Collector Negative

Remarks:

1. **Coupling ICFD to EM.** Only the resistive heating solver is currently available when coupling the ICFD solver with the EM solver (see *EM_CONTROL).

*EM_MAT_002

Purpose: Define an electromagnetic material type and properties whose permeability is different than the free space permeability.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MTYPE	SIGMA	EOSID	MUREL	EOSMU	DEATHT	
Type	A	I	F	I	F	I	F	
Default	none	none	none	none	none	none	10^{28}	

Optional card

Card 2	1	2	3	4	5	6	7	8
Variable		EOSID2						
Type		I						
Default		none						

VARIABLE	DESCRIPTION
MID	Material identification. A unique number or label must be specified (see *PART).
MTYPE	Electromagnetism type of the material: EQ.0: Air or vacuum EQ.1: Insulator material. These materials have the same electromagnetism behavior as MTYPE = 0. EQ.2: Conductor carrying a source. In these conductors, the eddy current problem is solved, which gives the actual current density. Typically, this would correspond to the coil. EQ.4: Conductor not connected to any current or voltage source, where the Eddy current problem is solved. Typically, this would correspond to the workpiece.

VARIABLE	DESCRIPTION
SIGMA	Initial electrical conductivity of the material
EOSID	ID of the EOS to be used for the electrical conductivity (see *EM_-EOS cards)
MUREL	Relative permeability which is the ratio of the permeability of a specific medium to the permeability of free space ($\mu_r = \mu/\mu_0$)
EOSMU	ID of the EOS to be used to define the nonlinear behavior of μ. Note: if EOSMU is defined, MUREL will be used for the initial value only. See EM_EOS_PERMEABILITY.
DEATHT	Death time for the material. After DEATHT, the material will no longer be considered a conductor and will be removed from the EM solve. If a negative value is entered, a *DEFINE_FUNCTION will be expected. The following parameters are allowed: (vx, vy, vz, temp, vol, mass, Ex, Ey, Ez, Bx, By, Bz, Fx, Fy, Fz, JHrate, time). Fx, Fy, and Fz refer to the components of the Lorentz force vector. A negative value returned by the *DEFINE_FUNCTION corresponds to a 'dead' or inactive element. Once an element has been removed from the EM solve, it cannot return.
EOSID2	Optional ID of the EOS for specifying the behavior of μ by an equation of state. See *EM_EOS_TABULATED1 and *EM_EOS_TABULATED2.

*EM_MAT_003

Purpose: Define an electromagnetic material type whose electromagnetic conductivity is defined by a (3 × 3) tensor matrix. Applications include composite materials. This material only applies to solid elements.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MTYPE	SIGMA11	SIGMA22	SIGMA33	BETA	CM	
Type	I	I	F	F	F	F	F	

Card 2	1	2	3	4	5	6	7	8
Variable	SIGMA12	SIGMA13	SIGMA21	SIGMA23	SIGMA31	SIGMA32	AOPT	LAMBDA
Type	I	I	F	F	F	F	I	F

Card 3	1	2	3	4	5	6	7	8
Variable	XP	YP	ZP	A1	A2	A3	MACF	
Type	F	F	F	F	F	F	I	

Card 4	1	2	3	4	5	6	7	8
Variable	V1	V2	V3	D1	D2	D3		
Type	F	F	F	F	F	F		

VARIABLE	DESCRIPTION
MID	Material ID. A unique number must be specified (see *PART).
MTYPE	Defines the electromagnetism type of the material: EQ.0: Air or vacuum

VARIABLE	DESCRIPTION
	EQ.1: Insulator material. These materials have the same electromagnetism behavior as MTYPE = 0.
	EQ.2: Conductor carrying a source. For these conductors, the EM solver solves the eddy current problem, which gives the actual current density. Typically, this type would correspond to the coil. In electrophysiology (EP), it corresponds to the tissue where the monodomain equations are solved for EMSOL = 11 or EMSOL = 13. For this case, an *EM_EP_CELLMODEL must be associated to this *EM_MAT_003.
	EQ.4: Conductor not connected to any current or voltage source, where the Eddy current problem is solved. Typically, this type would correspond to the workpiece. In electrophysiology (EP), for EMSOL = 11, 12 or 13, it corresponds to the bath surrounding the tissue for which only the external potential is found. In the case of electrophysiology, no *EM_EP_CELLMODEL should be associated with this material.
SIGMA11	The 1, 1 term in the 3 × 3 electromagnetic conductivity tensor matrix. Note that 1 corresponds to the **a** material direction. LT.0.0: |SIGMA11| corresponds to the ID of a *DEFINE_FUNCTION. See Remark 2 for available parameters.
SIGMA12	The 1, 2 term in the 3 × 3 electromagnetic conductivity tensor matrix. Note that 2 corresponds to the **b** material direction. LT.0.0: |SIGMA12| corresponds to the ID of a *DEFINE_FUNCTION. See Remark 2 for available parameters.
⋮	⋮
SIGMA33	The 3, 3 term in the 3 × 3 electromagnetic conductivity tensor matrix. LT.0.0: |SIGMA33| corresponds to the ID of a *DEFINE_FUNCTION. See Remark 2 for available parameters.
BETA	Surface to volume ratio of the cell membrane (to be used only when EMSOL = 11 or 12 in *EM_CONTROL).
CM	Membrane capacitance per unit area (to be used only when EMSOL = 11 or 12 in EM_CONTROL).

VARIABLE	DESCRIPTION
AOPT	Material axes option (see *MAT_002 for a more detailed description): EQ.0.0: Locally orthotropic with material axes determined by element nodes. The **a**-direction is from node 1 to node 2 of the element. The **b**-direction is orthogonal to the **a**-direction and is in the plane formed by nodes 1, 2, and 4. EQ.1.0: Locally orthotropic with material axes determined by a point in space, *P*, and the global location of the element center; this is the **a**-direction. EQ.2.0: Globally orthotropic with material axes determined by vectors defined below, as with *DEFINE_COORDINATE_VECTOR. EQ.3.0: Locally orthotropic material axes determined by a vector **v** and the normal vector to the plane of the element.The plane of a solid element is the midsurface between the inner surface and outer surface defined by the first four nodes and the last four nodes of the connectivity of the element, respectively. Thus, for solid elements, AOPT = 3 is only available for hexahedrons. **a** is determined by taking the cross product of **v** with the normal vector, **b** is determined by taking the cross product of the normal vector with **a**, and **c** is the normal vector. Then **a** and **b** are rotated about **c** by an angle BETA. BETA may be set in the keyword input for the element. EQ.4.0: Locally orthotropic in cylindrical coordinate system with the material axes determined by a vector, **v**, and an originating point, *P*, which define the centerline axis.
XP, YP, ZP	Coordinates of point, *P*, for AOPT = 1 and 4
A1, A2, A3	Components of vector, **a**, for AOPT = 2
MACF	Material axes change flag for solid elements: EQ.1: No change, default
V1, V2, V3	Components of vector, **v**, for AOPT = 3 and 4.
D1, D2, D3	Components of vector, **d**, for AOPT = 2

VARIABLE	DESCRIPTION
LAMBDA	Intra- to extracellular conductivity ratio. When non-empty, the elliptic equation is solved to compute extracellular potentials (to be used only when EMSOL = 11 in *EM_CONTROL).

Remarks:

1. **Material Directions.** See the manual page for *MAT_002 for a description of how the principal material directions, {**a**, **b**, **c**}, are determined with AOPT. Note that *EM_MAT_003 only works for solid elements. The AOPT options illustrated in the AOPT figure of *MAT_002 can define the material directions for all elements of the parts that use the material.

2. **Function Parameters.** The available parameters for the *DEFINE_FUNCTIONs are: f(time, emdt, x_ele, y_ele, z_ele, ieleuser). Here, time is the current EM time; emdt is the current EM time step; x_ele, y_ele, and z_ele are the location of the element; and ieleuser is the element ID.

*EM_MAT_004

Purpose: Define the electromagnetic material type and properties for conducting shells in a 3D problem or in a 2D resistive heating problem.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MTYPE	SIGMA	EOSID	NELE	MUREL	EOSMU	DEATHT
Type	I	I	F	I	I	F	I	F
Default	none	none	none	none	1	1.	none	10^{28}

VARIABLE	DESCRIPTION
MID	Material ID: refers to MID in the *PART card.
MTYPE	Defines the electromagnetism type of the material: EQ.0: Air or vacuum EQ.1: Insulator material. These materials have the same electromagnetism behavior as MTYPE = 0. EQ.2: Conductor carrying a source. In these conductors, the eddy current problem is solved, which gives the actual current density. Typically, this would correspond to the coil. EQ.4: Conductor not connected to any current or voltage source, where the Eddy current problem is solved. Typically, this would correspond to the workpiece.
SIGMA	Initial electrical conductivity of the material
EOSID	ID of the EOS to be used for the electrical conductivity (see *EM_-EOS cards).
NELE	Number of elements in the thickness of the shell. Note that you must make sure your mesh fine enough to correctly capture the inductive-diffusive effects (see skin depth definition).
MUREL	Relative permeability which is the ratio of the permeability of a specific medium to the permeability of free space ($\mu_r = \mu/\mu_0$).

VARIABLE	DESCRIPTION
EOSMU	ID of the EOS to be used to define the nonlinear behavior of μ. Note that if EOSMU is defined, MUREL will be used for the initial value only. See *EM_EOS_PERMEABILITY.
DEATHT	Death time for the material. After DEATHT, the material will no longer be considered a conductor and will be removed from the EM solve. If a negative value is entered, a *DEFINE_FUNCTION will be expected. The following parameters are allowed: (vx, vy, vz, temp, vol, mass, Ex, Ey, Ez, Bx, By, Bz, Fx, Fy, Fz, JHrate, time). Fx, Fy, and Fz refer to the components of the Lorentz force vector. A negative value returned by the *DEFINE_FUNCTION corresponds to a 'dead' or inactive element. Once an element has been removed from the EM solve, it cannot return.

*EM_MAT_005

Purpose: Define an electromagnetic material for which two material conductivities are specified per EM node and electromagnetic conductivities are defined by a (3 × 3) tensor matrix. Applications of this material include the Randles Batmac model (see Remark 1) and the electrophysiology bidomain model. This material is only supported for solid elements.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MTYPE	SIGMAXXA	SIGMAYYA	SIGMAZZA			
Type	I	I	F	F	F			

Card 2	1	2	3	4	5	6	7	8
Variable	SIGMAXYA	SIGMAXZA	SIGMAYXA	SIGMAYZA	SIGMAZXA	SIGMAZYA		
Type	F	F	F	F	F	F		

Card 3	1	2	3	4	5	6	7	8
Variable			SIGMAXXB	SIGMAYYB	SIGMAZZB			
Type			F	F	F			

Card 4	1	2	3	4	5	6	7	8
Variable	SIGMAXYB	SIGMAXZB	SIGMAYXB	SIGMAYZB	SIGMAZXB	SIGMAZYB		
Type	F	F	F	F	F	F		

Card 5	1	2	3	4	5	6	7	8
Variable	AOPT	XP	YP	ZP	A1	A2	A3	MACF
Type	F	F	F	F	F	F	I	I

Card 6	1	2	3	4	5	6	7	8
Variable	V1	V2	V3	D1	D2	D3		
Type	F	F	F	F	F	F		

VARIABLE	DESCRIPTION
MID	Material ID. A unique number must be specified (see *PART).
MTYPE	Defines the electromagnetism type of the material: EQ.0: Air or vacuum EQ.1: Insulator material. These materials have the same electromagnetism behavior as MTYPE = 0. EQ.2: In electrophysiology, it corresponds to the tissue, where the bidomain equations will be solved for EMSOL = 12 or EMSOL = 13. An *EM_EP_CELLMODEL must be associated with this material. EQ.4: In electrophysiology, it corresponds to the bath where only the external potential is solved for. No *EM_EP_CELLMODEL should be associated with this material. EQ.5: Material associated with *EM_RANDLES_BATMAC. See Remark 1.
SIGMAXXA/B	The 1, 1 term in the 3 × 3 electromagnetic conductivity tensor matrix for the two conductivities. For the BatMac model, A is for the potential on the positive current collector, and B is for the potential on the negative current collector. For the bidomain model in Electrophysiology, A is for the intracellular potential, and B is for the extracellular potential. Note that 1 corresponds to the **a** material direction. LT.0.0: |SIGMAXXA/B| corresponds to the ID of a *DEFINE_FUNCTION. See Remark 3 for available parameters.

SIGMAXYA/B — The 1, 2 term in the 3 × 3 electromagnetic conductivity tensor matrix for the two conductivities. Note that 2 corresponds to the **b** material direction.

LT.0.0: |SIGMAXYA/B| corresponds to the ID of a *DEFINE_FUNCTION. See Remark 3 for available parameters.

⋮ ⋮

SIGMAZZA/B — The 3, 3 term in the 3 × 3 electromagnetic conductivity tensor matrix for the two conductivities. Note that 3 corresponds to the **c** material direction.

LT.0.0: |SIGMAZZA/B| corresponds to the ID of a *DEFINE_FUNCTION. See Remark 3 for available parameters.

AOPT — Material axes option (see *MAT_002 for a more detailed description):

EQ.0.0: Locally orthotropic with material axes determined by element nodes. The **a**-direction is from node 1 to node 2 of the element. The **b**-direction is orthogonal to the **a**-direction and is in the plane formed by nodes 1, 2, and 4.

EQ.1.0: Locally orthotropic with material axes determined by a point in space, P, and the global location of the element center; this is the **a**-direction.

EQ.2.0: Globally orthotropic with material axes determined by vectors defined below, as with *DEFINE_COORDINATE_VECTOR.

EQ.3.0: Locally orthotropic material axes determined by a vector **v** and the normal vector to the plane of the element.The plane of a solid element is the midsurface between the inner surface and outer surface defined by the first four nodes and the last four nodes of the connectivity of the element, respectively. Thus, for solid elements, AOPT = 3 is only available for hexahedrons. **a** is determined by taking the cross product of **v** with the normal vector, **b** is determined by taking the cross product of the normal vector with **a**, and **c** is the normal vector. Then **a** and **b** are rotated about **c** by an angle BETA. BETA may be set in the keyword input for the element.

EQ.4.0: Locally orthotropic in cylindrical coordinate system with the material axes determined by a vector, **v**, and an originating point, *P*, which define the centerline axis.

XP, YP, ZP — Define coordinates of point **p** for AOPT = 1 and 4.

A1, A2, A3 — Define components of vector **a** for AOPT = 2.

MACF — Material axes change flag for solid elements:

EQ.1: No change, default

V1, V2, V3 — Define components of vector **v** for AOPT = 3 and 4.

D1, D2, D3 — Define components of vector **d** for AOPT = 2.

Remarks:

1. **Using this Material with BatMac.** When this material is used in conjunction with the battery BatMac model, then the conductivities must be adjusted by the current collector thickness ratio over the total thickness of the cell. For example, if n_p is the number of positive current collectors, t_p the thickness of each individual positive current collector and Th the total thickness of the cell, then the conductivity for the positive current collector must be scaled by:

$$\frac{n_p \times t_p}{\text{Th}} .$$

2. **Material Directions.** See the manual page for *MAT_002 for a description of how the principal material directions, {**a**, **b**, **c**}, are determined with AOPT. Note that *EM_MAT_005 only works for solid elements. The AOPT options illustrated in the AOPT figure of *MAT_002 can define the material directions for all elements of the parts that use the material.

3. **Function Parameters.** The available parameters for the *DEFINE_FUNCTIONs are: f(time, emdt, x_ele, y_ele, z_ele, ieleuser). Here, time is the current EM time; emdt is the current EM time step; x_ele, y_ele, and z_ele are the location of the element; and ieleuser is the element ID.

*EM_MAT_006

Purpose: Define two conductivities per EM node for special applications (Randles Batmac).

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	MTYPE	SIGP	EOSP	SIGN	EOSN	DEATHT	
Type	I	I	F	I	F	I	F	
Default	none	none	none	none	none	none	10^{28}	

VARIABLE	DESCRIPTION
MID	Material ID: refers to MID in the *PART card.
MTYPE	Defines the electromagnetism type of the material: EQ.0: Air or vacuum EQ.1: Insulator material. These materials have the same electromagnetism behavior as MTYPE = 0. EQ.5: Material associated to *EM_RANDLES_BATMAC
SIGP/SIGN	Conductivities of the positive / negative current collector materials
EOSP/EOSN	Optional ID of the EOS to be used for the two conductivities
DEATHT	Death time for the material. After DEATHT, the material will no longer be considered a conductor and removed from the EM solve. If a negative value is entered, a |DEATHT| is a *DEFINE_FUNCTION ID. The following parameters are allowed: (vx, vy, vz, temp, vol, mass, Ex, Ey, Ez, Bx, By, Bz, Fx, Fy, Fz, JHrate, time). The vector (Fx, Fy, Fz) refers to the Lorentz force vector. A negative value returned by the *DEFINE_FUNCTION corresponds to a "dead" or inactive element. Once an element has been removed from the EM solve, it cannot return.

Remarks:

1. When this material is used in conjunction with the battery BatMac model, then the conductivities must be adjusted by the current collector thickness ratio over the total thickness of the cell. For example, if n_p is the number of positive current collectors, t_p is the thickness of each individual positive current collector and Th is the total thickness of the cell, then the conductivity for the positive current collector must be scaled by: $n_p \times t_p/\text{Th}$.

*EM_OUTPUT

Purpose: Define the level of EM related output on the screen and in the messag file.

Card 1	1	2	3	4	5	6	7	8
Variable	MATS	MATF	SOLS	SOLF	MESH	MEM	TIMING	
Type	I	I	I	I	I	I	I	
Default	0	0	0	0	0	0	0	

VARIABLE	DESCRIPTION
MATS	Level of matrix assembly output to the screen: EQ.0: no output EQ.1: basic assembly steps EQ.2: basic assembly steps + percentage completed + final statistics EQ.3: basic assembly steps + percentage completed + statistics at each percentage of completion
MATF	Level of matrix assembly output to the messag file: EQ.0: no output EQ.1: basic assembly steps EQ.2: basic assembly steps + percentage completed + final statistics EQ.3: basic assembly steps + percentage completed + statistics at each percentage of completion
SOLS	Level of solver output on the screen: EQ.0: no output EQ.1: global information at each FEM iteration EQ.2: detailed information at each FEM iteration
SOLF	Level of solver output to the messag file: EQ.0: no output

VARIABLE	DESCRIPTION
	EQ.1: global information at each FEM iteration
	EQ.2: detailed information at each FEM iteration
MESH	Controls the output of the mesh data to the d3hsp file:
	EQ.0: no mesh output written.
	EQ.1: mesh info written.
MEMORY	Controls the output of information about the memory used by the EM solve to the messag file:
	EQ.0: no memory information written.
	EQ.1: memory information written.
TIMING	Controls the output of information about the time spent in the different parts of the EM solver to the messag file
	EQ.0: no timing information written.
	EQ.1: timing information written.

*EM_OUTPUT_VTK

Purpose: Cause the output of vtk files. It can be used in applications that require two material conductivities per EM node and whose electromagnetic conductivities are defined by a (3 × 3) tensor matrix. These applications include the Randles Batmac model and the Electrophysiology Bidomain model.

Card 1	1	2	3	4	5	6	7	8
Variable	VTKTYPE	VTKT						
Type	I	F						

VARIABLE	DESCRIPTION
VTKTYPE	Type of the vtk files output. EQ.1: A single .vtk file EQ.2: Parallel unstructured points data (.pvtu files), recommended in mpp executions.
VTKT	Time period at which vtk files are exported

*EM_PERMANENT_MAGNET

Purpose: Defines a permanent magnet.

Card Summary:

Card 1. This card is required.

ID	PID	MTYPE	NORTH	SOUTH	HC		

Card 2a. This card is included if MTYPE = 3.

X	Y	Z					

Card 2b. This card is included if MTYPE = 4.

NID1	NID2						

Card 2c. This card is included if MTYPE = 5 or 6.

X	Y	Z	NDIVIS	AXIS	DIR/X2	Y2	Z2

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	ID	PID	MTYPE	NORTH	SOUTH	HC		
Type	I	I	I	I	I	I		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
ID	ID of the magnet
PID	Part ID
MTYPE	Magnet definition type: EQ.0: Magnet defined by two node sets for North and South Poles.

VARIABLE	DESCRIPTION
	EQ.1: Magnet defined by two segments sets for North and South Poles. EQ.3: Magnet defined by a global vector orientation. EQ.4: Magnet defined by a global vector orientation given by two node IDs EQ.5: Magnetic gear with NDIVIS magnets oriented around an axis given by AXIS and centered at point (X, Y, Z). The pole orientations alternate for each magnet in the gear, meaning they alternate at increments of 360/NDIVIS around the gear starting along the vector given by (X2, Y2, Z2). See Figure 6-4. EQ.6: Magnetic gear with NDIVIS magnets oriented around an axis given by AXIS and centered at point (X, Y, Z). The pole orientations alternate for each magnet in the gear, meaning they alternate at increments of 360/NDIVIS around the gear starting along the vector given by DIR. See Figure 6-4.
NORTH	Set ID of the magnet north face for MTYPE = 0 and 1
SOUTH	Set ID of the magnet south face for MTYPE = 0 and 1
HC	Coercive force, H_c. See Remark 1. LT.0.0: |HC| refers to a load curve ID giving the coercive force as a function of time.

MTYPE = 3 Card. This card is only included for MTYPE = 3.

Card 2a	1	2	3	4	5	6	7	8
Variable	X	Y	Z					
Type	F	F	F					
Default	0.	0.	0.					

VARIABLE	DESCRIPTION
X, Y, Z	Orientation of magnetization vector

MTYPE = 4 Card. This card is only included for MTYPE = 4.

Card 2b	1	2	3	4	5	6	7	8
Variable	NID1	NID2						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
NID1 / NID2	Two node IDs defining the magnetization vector

Magnetic Gear Card. This card is only included if MTYPE = 5 or 6. See Figure 6-4.

Card 2c	1	2	3	4	5	6	7	8
Variable	X	Y	Z	NDIVIS	AXIS	DIR/X2	Y2	Z2
Type	F	F	F	I	I	I/F	F	F
Default	0.	0.	0.	none	none	none	none	none

VARIABLE	DESCRIPTION
X, Y, Z	Origin / center point of the magnetic gear
NDIVIS	Number of subdivisions, that is, number of magnets around the circle
AXIS	Normal vector to the magnets: EQ.1: Global *X* axis EQ.2: Global *Y* axis EQ.3: Global *Z* axis
DIR	Directional vector giving the location of the starting magnet / starting magnetic orientation if MTYPE = 6: EQ.1: Global *X* axis

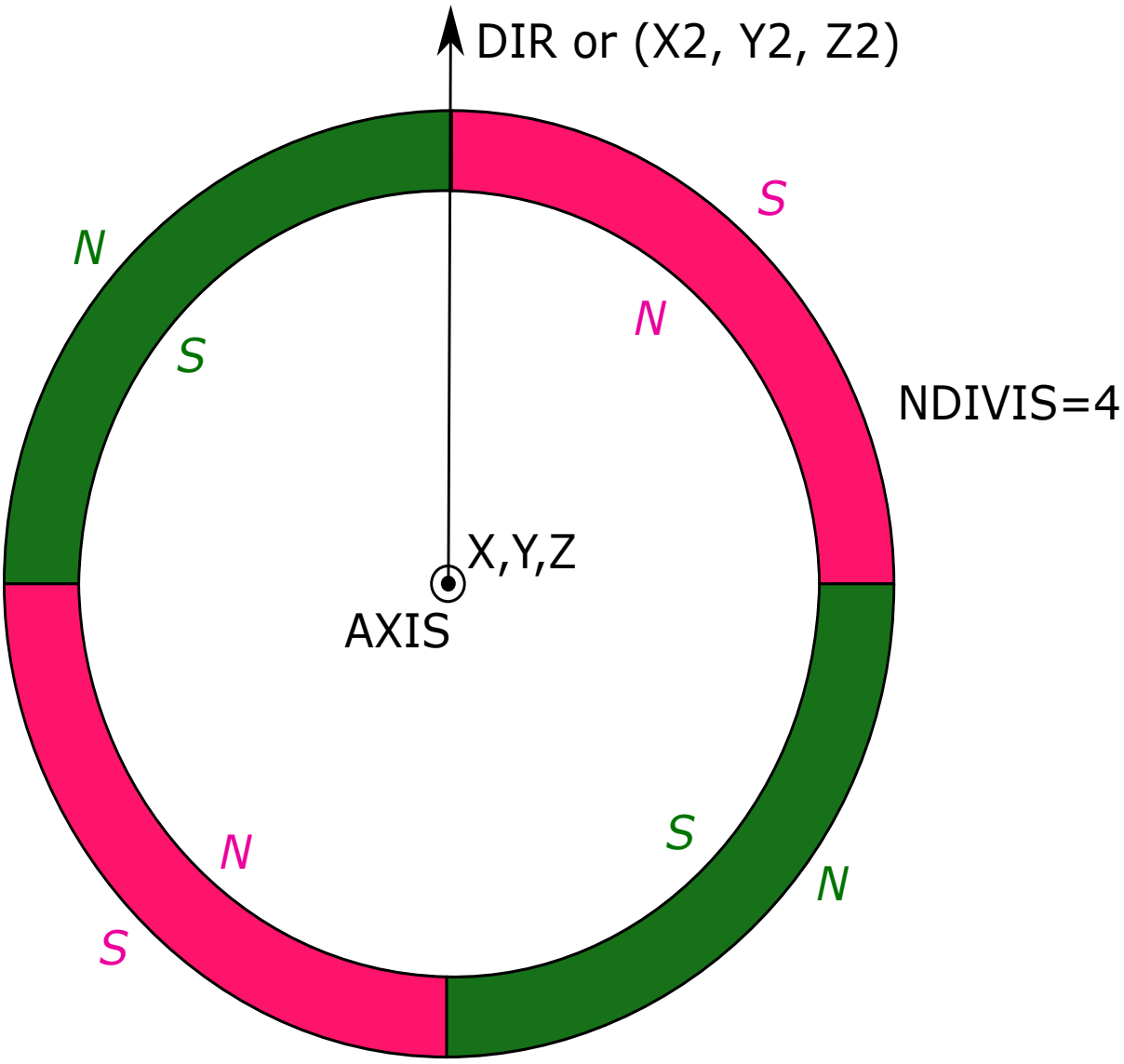


Figure 6-4. Example of Magnetic Gear (MTYPE = 5 or 6)

VARIABLE	DESCRIPTION
	EQ.2: Global Y axis
	EQ.3: Global Z axis
X2, Y2, Z2	Directional vector coordinates giving the starting magnet / starting magnetic orientation if MTYPE = 5

Remark:

1. **Coercive Force.** The absolute value of coercive force H_c applied to the magnet (A/m) relates to the Residual induction B_r by the following relation:

$$H_c = B_r/\mu$$

Here, μ is the magnet's permeability defined in *EM_MAT_002 using a constant relative permeability $\mu = \mu_r\mu_0$.

The coercive force can also be expressed using the energy product $BH_{\max}$:

$$H_c = 2\sqrt{\frac{BH_{\max}}{\mu}}$$

*EM_POINT_SET

Purpose: This keyword creates a set of points which can be used by the *EM_DATABASE_POINTOUT keyword.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	PSID	PSTYPE	VX	VY	VZ			
Type	I	I	F	F	F			
Default	0	0	0.	0.	0.			

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	PID	X	Y	Z	POS			
Type	I	F	F	F	I			
Default	none	none	none	none	0			

VARIABLE	DESCRIPTION
PSID	Point Set ID.
PSTYPE	Point Set type : EQ.0: Fixed points. EQ.1: Tracer points using prescribed velocity.
VX, VY, VZ	Constant velocities to be used when PSTYPE = 1
PID	Point ID
X, Y, Z	Point initial coordinates

VARIABLE	DESCRIPTION
POS	Position flag (for 2D see Remark 1) : EQ.0 (default) : The solver determines if the point is inside or outside of the conductors. EQ.1: Point outside of the conductors during the entire simulation. The solver does not check; hence a gain in computation time.

Remarks:

1. If using *EM_2DAXI notice that the conductors represents the corresponding 3D conductors.

*EM_RANDLES_BATMAC

Purpose: define the distributed Randles circuit parameters for a Randles cell when using the batmac model. The batmac model is a macro battery model where solid elements are retained for the solid mechanics and thermal solve and where each conducting node will have its own Randles circuit associated to it. Must be used with *EM_MAT_006 or *EM_-MAT_005.

Card 1	1	2	3	4	5	6	7	8
Variable	RDLID	RDLTYPE	RDLAREA	PSID				
Type	I	I	I	I				
Default	none	none	none	none				

Card 2	1	2	3	4	5	6	7	8
Variable	Q	CQ	SOCINIT	SOCTOU				
Type	F	F	F	F				
Default	none	none	none	none				

Card 3.a	1	2	3	4	5	6	7	8
Variable	R0CHA	R0DIS	R10CHA	R10DIS	C10CHA	C10DIS		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Optional Card. This card only needs to be defined for RDLTYPE greater than 1.

Card 3.b	1	2	3	4	5	6	7	8
Variable	R20CHA	R20DIS	C20CHA	C20DIS	R30CHA	R30DIS	C30CHA	C30DIS
Type	F	F	F	F	F	F	F	F
Default	0.	0.	0.	0.	0.	0.	0.	0.

Optional thermal card.

Card 4	1	2	3	4	5	6	7	8
Variable	TEMP	FRTHER	ROTOTH	DUDT	TEMPU			
Type	F	I	I	F	I			
Default	0.	0	0	none	0			

Optional SOC shift card.

Card 5	1	2	3	4	5	6	7	8
Variable	USESOCS	TAU	FLCID					
Type	I	F	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
RDLID	Id of the Randles Cell

VARIABLE	DESCRIPTION
RDLTYPE	Type of Randles Cell: EQ.-1: User defined equivalent circuit model. See Remark 3. EQ.0: 0-order Randles Cell EQ.1: 1-order Randles Cell EQ.2: 2-order Randles Cell EQ.3: 3-order Randles Cell
PSID	Part Set ID of all the parts composing the cell
RDLAREA	Randles Area: EQ.1: The parameters are per unit area and will be scaled in each Randle circuit by a factor depending on the local area of the circuit. Unit consistency in S.I : Ohms times square meters. EQ.2: Default. The parameters are defined for the whole cell and will be scaled in each Randle circuit by a factor depending on the local area of the circuit and the global area of the cell. Unit consistency in S.I : Ohms. EQ.3: The parameters are not scaled by area factors. Unit consistency in S.I : Ohms.
Q	Cell capacity.
CQ	SOC conversion factor (%/s), known to be equal to 1/36 in S.I units.
SOCINIT	Initial state of charge of the cell.
SOCTOU	Equilibrium voltage (OCV): GE.0.0: constant value LT.0.0: |SOCTOU| is a load curve ID defining equilibrium voltage (OCV) as a function of the state of charge (SOC).

VARIABLE	DESCRIPTION
R0CHA/ R10CHA/ C10CHA	$r_0/r_{10}/c_{10}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R0DIS/ R10DIS/ C10DIS	$r_0/r_{10}/c_{10}$ when the current flows in the discharge direction: GE.0.0: constant value. LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20CHA/ R30CHA/ C20CHA/ C30CHA	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the charge direction: GE.0.0: constant value. LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20DIS/ R30DIS/ C20DIS/ C30DIS	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the discharge direction: GE.0.0: constant value. LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
TEMP	Constant temperature value used for the Randles circuit parameters in case there is no coupling with the thermal solver (FRTHER = 0)
FRTHER	From Thermal : EQ.0: The temperature used in the Randles circuit parameters is TEMP EQ.1: The temperature used in the Randles circuit parameter is the temperature from the thermal solver.

VARIABLE	DESCRIPTION
R0TOTH	r_0 to Thermal: EQ.0: The joule heating in the resistance r0 is not added to the thermal solver EQ.1: The joule heating in the resistance r0 is added to the thermal solver
DUDT	If negative integer, load curve ID of the reversible heat as a function of SOC.
TEMPU	Temperature Unit : EQ.0: The temperature is in Celsius EQ.1: The Temperature is in Kelvin
USESOCS	Use SOC shift (See Remark 2) : EQ.0: Don't use the added SOCshift EQ.1: Use the added SOCshift
TAU	Damping time in the SOCshift equation.
FLCID	Load curve giving f(i) where I is the total current in the unit cell

Remarks:

1. **Model combinations.** The batmac model cannot be mixed with the solid or thick shell Randles models. It can however be used in conjunction with the meshless model.

2. **Accounting for Diffusion Limitations.** Sometimes, an extra term called SOCshift (or SocS) can be added at high rate discharges to account for diffusion limitations. The SOCshift is added to SOC for the calculation of the OCV $u(\text{SOC} + \text{SOCshift})$ and $r_0(\text{SOC} + \text{SOCshift})$. SOCshift satisfies the following equation:

$$\frac{d(\text{SOCshift})}{dt} + \frac{\text{SOCshift}}{\tau} = \frac{f(i(t))}{\tau}$$

with $\text{SOCshift}(t = 0) = 0$.

3. **User defined ECMs**. Randles circuits are based on the finite element Robin boundary condition, acting similarly to a convection boundary condition for the heat equation. As such, the circuit equation is decomposed into a term that will enter the left-hand-side of the stiffness matrix and a term that will be added to the right-hand-side. Changing the definition of those terms allows the user to replace the Randles circuits by any type of Equivalent Circuit model (ECM). This is the purpose of RDLTYPE = -1 where the term entering the l.h.s is defined by a negative integer referring to a *DEFINE_FUNCTION ID in R0CHA (unit consistency: Resistance) while a negative integer in R0DIS, associated to a *DEFINE_FUNCTION will give the term entering in the r.h.s (unit consistency : current).

4. **DEFINE FUNCTION** variables available in EM_RANDLES keywords :

*DEFINE_FUNCTIONs: / *Variable names* :	Randles Circuit parameters (r_0, r_{10}, c_{10} *etc*)	RDLTYPE = -1	Internal Short	Exothermic reaction
'time' : Current EM time	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'emdt' : Current EM timestep	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_sep,y_sep,z_sep' : Positive Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_sen,y_sen,z_sen' : Negative Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_ccp,y_ccp,z_ccp' : Positive Current collector coordinates	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_ccn,y_ccn,z_ccn' : Negative Current collector coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'pres' : Local pressure	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'rho' : Local density	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'vmstress' : Local von Mises stress	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'cond' : Local electrical conductivity	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'temp' : Local Temperature	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'tempRand' : Temperature associated to Randles Circuit	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'efstrain' : Local Effective strain	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'strainLocX/Y/Z' : Local strain in the X/Y/Z directions	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models

'soc,soceff' : Local state of charge, effective state of charge	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'current' : Transverse Randles current	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ocv,vc,volt,r0' : open charge voltage, damping voltage, total voltage, r0 resistance.	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'H_ex' : Exothermal heating power integrated over time (=exothermal heating energy) when exothermic reaction keyword is present.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'short' : Short gives the state of the Randles circuit. If short = 0, then the circuit is not shorted, if short = 1, then it is shorted.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ero : Ero gives the state of the erosion. If ero = 1, then the circuit is adjacent to an element which has been eroded, ero = 0 otherwise.	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models
'areaCircuit,areaCell, *areashortGlob'* : local Randles circuit area, Total Randles Cell area, Total Cell Shorted area.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'save1,save2,save3…,save10' : ten local variables that the user can define and that will be saved during the run and associated to each local Randles circuits.	No	Yes	No	No

*EM_RANDLES_EXOTHERMIC_REACTION

Purpose: This keyword allows the user to add an extra heat source term to the Randles circuit nodes in order to account for thermal runaway situations.

Card 1	1	2	3	4	5	6	7	8
Variable	AREATYPE	FUNCTID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
AREATYPE	Works the same way as RDLAREA in *EM_RANDLES_SOLID or in *EM_RANDLES_TSHELL : EQ.1: The heat source in FUNCTID is per unit area so that, for each local Randles circuit, the result returned by FUNCTID is multiplied by a factor areaLoc (areaLoc is the local area associated to each Randles circuit while areaGlob is the area of the whole cell) (W.m^2). EQ.2: Default. The heat source in FUNCTID is for the whole cell (the whole cell is shorted), so that, for each Randles circuit, the result returned by FUNCTID is multiplied by a factor areaLoc/areaGlob (W) . EQ.3: The heat source returned by FUNCTID is taken as is in each Randles circuit (W).
FUNCTID	DEFINE_FUNCTION ID giving the local heat source function of local parameters for the local Randles circuit. See Remark 1.

Remarks:

1. **DEFINE FUNCTION** variables available in EM_RANDLES keywords :

*DEFINE_FUNCTIONs: / *Variable names* :	Randles Circuit parameters (r_0, r_{10}, c_{10} *etc*)	RDLTYPE = -1	Internal Short	Exothermic reaction

'time' : Current EM time	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'emdt : Current EM timestep	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_sep,y_sep,z_sep' : Positive Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_sen,y_sen,z_sen' : Negative Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_ccp,y_ccp,z_ccp' : Positive Current collector coordinates	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_ccn,y_ccn,z_ccn' : Negative Current collector coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'pres' : Local pressure	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'rho' : Local density	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'vmstress' : Local von Mises stress	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'cond' : Local electrical conductivity	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'temp' : Local Temperature	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'tempRand' : Temperature associated to Randles Circuit	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'efstrain' : Local Effective strain	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'strainLocX/Y/Z' : Local strain in the X/Y/Z directions	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'soc,soceff' : Local state of charge, effective state of charge	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'current' : Transverse Randles current	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ocv,vc,volt,r0' : open charge voltage, damping voltage, total voltage, r0 resistance.	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models

'H_ex' : Exothermal heating power integrated over time (=exothermal heating energy) when exothermic reaction keyword is present.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'short' : Short gives the state of the Randles circuit. If short = 0, then the circuit is not shorted, if short = 1, then it is shorted.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ero : Ero gives the state of the erosion. If ero = 1, then the circuit is adjacent to an element which has been eroded, ero = 0 otherwise.	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models
'areaCircuit,areaCell, *areashortGlob'* : local Randles circuit area, Total Randles Cell area, Total Cell Shorted area.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'save1,save2,save3…,save10' : ten local variables that the user can define and that will be saved during the run and associated to each local Randles circuits.	No	Yes	No	No

*EM_RANDLES_MESHLESS

Purpose: define the distributed Randles circuit parameters for a Randles cell which is not associated with a mesh (lumped Randles circuit).

Card 1	1	2	3	4	5	6	7	8
Variable	RDLID	RDLTYPE						
Type	I	I						
Default	none	none						

Card 2	1	2	3	4	5	6	7	8
Variable	Q	CQ	SOCINIT	SOCTOU				
Type	F	F	F	F				
Default	none	none	none	none				

Card 3.a	1	2	3	4	5	6	7	8
Variable	R0CHA	R0DIS	R10CHA	R10DIS	C10CHA	C10DIS		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Optional Card. This card only needs to be defined for RDLTYPE greater than 1.

Card 3.b	1	2	3	4	5	6	7	8
Variable	R20CHA	R20DIS	C20CHA	C20DIS	R30CHA	R30DIS	C30CHA	C30DIS
Type	F	F	F	F	F	F	F	F
Default	0.	0.	0.	0.	0.	0.	0.	0.

Thermal Optional card.

Card 4	1	2	3	4	5	6	7	8
Variable	TEMP			DUDT	TEMPU			
Type	F			F	I			
Default	0.			none	0			

SOC shift Optional card.

Card 5	1	2	3	4	5	6	7	8
Variable	USESOCS	TAU	FLCID					
Type	I	F	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
RDLID	Id of the Randles Cell

VARIABLE	DESCRIPTION
RDLTYPE	Type of Randles Cell EQ.0: 0-order Randles Cell. EQ.1: 1-order Randles Cell. EQ.2: 2-order Randles Cell. EQ.3: 3-order Randles Cell.
Q	Cell capacity.
CQ	SOC conversion factor (%/s), known to be equal to 1/36 in S.I units.
SOCINIT	Initial state of charge of the cell.
SOCTOU	Equilibrium voltage (OCV): GE.0.0: constant value LT.0.0: |SOCTOU| is a load curve ID defining equilibrium voltage (OCV) as a function of the state of charge (SOC).
R0CHA/ R10CHA/ C10CHA	$r_0/r_{10}/c_{10}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 3 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R0DIS/ R10DIS/ C10DIS	$r_0/r_{10}/c_{10}$ when the current flows in the discharge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 3 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20CHA/ R30CHA/ C20CHA/ C30CHA	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.

VARIABLE	DESCRIPTION
R20DIS/ R30DIS/ C20DIS/ C30DIS	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the discharge direction: GE.0.0: constant value. LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
TEMP	Constant temperature value used for the Randles circuit parameters in case there is no coupling with the thermal solver.
DUDT	If negative integer, load curve ID of the reversible heat as a function of SOC.
TEMPU	Temperature Unit : EQ.0: The temperature is in Celsius EQ.1: The Temperature is in Kelvin
USESOCS	Use SOC shift (See Remark 1) : EQ.0: Don't use the added SOCshift EQ.1: Use the added SOCshift
TAU	Damping time in the SOCshift equation (See Remark 1)
FLCID	Load curve giving f(i) where I is the total current in the unit cell

Remarks:

1. **Accounting for Diffusion Limitations.** Sometimes, an extra term called SOCshift (or SocS) can be added at high rate discharges to account for diffusion limitations. The SOCshift is added to SOC for the calculation of the OCV $u(\text{SOC} + \text{SOCshift})$ and $r_0(\text{SOC} + \text{SOCshift})$. SOCshift satisfies the following equation:

$$\frac{d(\text{SOCshift})}{dt} + \frac{\text{SOCshift}}{\tau} = \frac{f(i(t))}{\tau}$$

with $\text{SOCshift}(t = 0) = 0$.

2. **User defined ECMs**. Randles circuits are based on the finite element Robin boundary condition, acting similarly to a convection boundary condition for the heat equation. As such, the circuit equation is decomposed into a term that will

enter the left-hand-side of the stiffness matrix and a term that will be added to the right-hand-side. Changing the definition of those terms allows the user to replace the Randles circuits by any type of Equivalent Circuit model (ECM). This is the purpose of RDLTYPE = -1 where the term entering the l.h.s is defined by a negative integer referring to a *DEFINE_FUNCTION ID in R0CHA (unit consistency: Resistance) while a negative integer in R0DIS, associated to a *DEFINE_FUNCTION will give the term entering in the r.h.s (unit consistency : current).

3. **DEFINE FUNCTION** variables available in EM_RANDLES keywords :

*DEFINE_FUNCTIONs: / *Variable names* :	Randles Circuit parameters (r_0, r_{10}, c_{10} *etc*)	RDLTYPE = -1	Internal Short	Exothermic reaction
'time' : Current EM time	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'emdt' : Current EM timestep	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_sep,y_sep,z_sep' : Positive Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_sen,y_sen,z_sen' : Negative Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_ccp,y_ccp,z_ccp' : Positive Current collector coordinates	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_ccn,y_ccn,z_ccn' : Negative Current collector coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'pres' : Local pressure	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'rho' : Local density	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'vmstress' : Local von Mises stress	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'cond' : Local electrical conductivity	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'temp' : Local Temperature	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models

'tempRand' : Temperature associated to Randles Circuit	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'efstrain' : Local Effective strain	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'strainLocX/Y/Z' : Local strain in the X/Y/Z directions	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'soc,soceff' : Local state of charge, effective state of charge	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'current' : Transverse Randles current	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ocv,vc,volt,r0' : open charge voltage, damping voltage, total voltage, r0 resistance.	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'H_ex' : Exothermal heating power integrated over time (=exothermal heating energy) when exothermic reaction keyword is present.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'short' : Short gives the state of the Randles circuit. If short = 0, then the circuit is not shorted, if short = 1, then it is shorted.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ero : Ero gives the state of the erosion. If ero = 1, then the circuit is adjacent to an element which has been eroded, ero = 0 otherwise.	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models
'areaCircuit,areaCell, *areashortGlob'* : local Randles circuit area, Total Randles Cell area, Total Cell Shorted area.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'save1,save2,save3…,save10' : ten local variables that the user can define and that will be saved during the run and associated to each local Randles circuits.	No	Yes	No	No

*EM_RANDLES_TSHELL

Purpose: Define the distributed Randles circuit parameters for a Randles cell when using a composite tshell mechanical model.

Card 1	1	2	3	4	5	6	7	8
Variable	RDLID	RDLTYPE	RDLAREA	PSID				
Type	I	I	I	I				
Default	none	none	2	none				

Card 2	1	2	3	4	5	6	7	8
Variable	Q	CQ	SOCINIT	SOCTOU				
Type	F	F	F	F				
Default	none	none	none	none				

Card 3.a	1	2	3	4	5	6	7	8
Variable	R0CHA	R0DIS	R10CHA	R10DIS	C10CHA	C10DIS		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Optional Card. This card only needs to be defined for RDLTYPE greater than 1.

Card 3.b	1	2	3	4	5	6	7	8
Variable	R20CHA	R20DIS	C20CHA	C20DIS	R30CHA	R30DIS	C30CHA	C30DIS
Type	F	F	F	F	F	F	F	F
Default	0.	0.	0.	0.	0.	0.	0.	0.

Optional Thermal Card.

Card 4	1	2	3	4	5	6	7	8
Variable	TEMP	FRTHER	ROTOTH	DUDT	TEMPU			
Type	F	I	I	F	I			
Default	0.	0	0	0.0	0			

Optional SOCShift Card.

Card 5	1	2	3	4	5	6	7	8
Variable	USESOCS	TAU	FLCID					
Type	I	F	I					
Default	0	0.0	0					

VARIABLE	DESCRIPTION
RDLID	ID of the Randles Cell

VARIABLE	DESCRIPTION
RDLTYPE	Type of Randles Cell: EQ.-1: User defined equivalent circuit model. See Remark 3. EQ.0: 0-order Randles Cell EQ.1: 1-order Randles Cell EQ.2: 2-order Randles Cell EQ.3: 3-order Randles Cell
PSID	Part Set ID of all the parts composing the cell
RDLAREA	Randles Area: EQ.1: the parameters are per unit area and will be scaled in each Randle circuit by a factor depending on the local area of the circuit. Unit consistency in S.I : Ohms times square meters. EQ.2: the parameters are defined for the whole cell and will be scaled in each Randle circuit by a factor depending on the local area of the circuit and the global area of the cell (default). Unit consistency in S.I : Ohms. EQ.3: the parameters are not scaled by area factors. Unit consistency in S.I : Ohms.
Q	Cell capacity
CQ	SOC conversion factor (%/s), known to be equal to 1/36 in SI units
SOCINIT	Initial state of charge of the cell
SOCTOU	Equilibrium voltage (OCV): GE.0.0: constant value LT.0.0: |SOCTOU| is a load curve ID defining equilibrium voltage (OCV) as a function of the state of charge (SOC).
R0CHA/ R10CHA/ C10CHA	$r_0/r_{10}/c_{10}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.

VARIABLE	DESCRIPTION
R0DIS/ R10DIS/ C10DIS	$r_0/r_{10}/c_{10}$ when the current flows in the discharge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20CHA/ R30CHA/ C20CHA/ C30CHA	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20DIS/ R30DIS/ C20DIS/ C30DIS	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the discharge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
TEMP	Constant temperature value used for the Randles circuit parameters in case there is no coupling with the thermal solver (FRTHER = 0)
FRTHER	From thermal: EQ.0: the temperature used in the Randles circuit parameters is TEMP EQ.1: the temperature used in the Randles circuit parameter is the temperature from the thermal solver.
R0TOTH	r_0 to thermal: EQ.0: the joule heating in the resistance r_0 is not added to the thermal solver. EQ.1: the joule heating in the resistance r_0 is added to the thermal solver.
DUDT	If negative integer, load curve ID of the reversible heat as a function of SOC.

VARIABLE	DESCRIPTION
TEMPU	Temperature unit: EQ.0: the temperature is in Celsius. EQ.1: the temperature is in Kelvin.
USESOCS	Use SOCshift (see Remark 2): EQ.0: don't use the added SOCshift. EQ.1: use the added SOCshift.
TAU	Damping time in the SOCshift equation (see Remark 2)
FLCID	Load curve giving $f(i)$ where i is the total current in the unit cell

Remarks:

1. **Sectioning of Circuit.** Each part of PSID is defined by *PART_COMPOSITE_-TSHELL. With this keyword for defining the part, each layer of a part can serve a different function, namely, as a current collector positive, current collector negative, separator, negative electrode, or positive electrode. A given layer's function is defined in the RDLTYPE field of *EM_MAT_001.

2. **Accounting for Diffusion Limitations.** Sometimes, an extra term called SOCshift (or SocS) can be added at high rate discharges to account for diffusion limitations. The SOCshift is added to SOC for the calculation of the OCV $u(\text{SOC} + \text{SOCshift})$ and $r_0(\text{SOC} + \text{SOCshift})$. SOCshift satisfies the following equation:

$$\frac{d(\text{SOCshift})}{dt} + \frac{\text{SOCshift}}{\tau} = \frac{f(i(t))}{\tau}$$

with $\text{SOCshift}(t = 0) = 0$.

3. **User defined ECMs**. Randles circuits are based on the finite element Robin boundary condition, acting similarly to a convection boundary condition for the heat equation. As such, the circuit equation is decomposed into a term that will enter the left-hand-side of the stiffness matrix and a term that will be added to the right-hand-side. Changing the definition of those terms allows the user to replace the Randles circuits by any type of Equivalent Circuit model (ECM). This is the purpose of RDLTYPE = -1 where the term entering the l.h.s is defined by a negative integer referring to a *DEFINE_FUNCTION ID in R0CHA (unit consistency: Resistance) while a negative integer in R0DIS, associated to a *DE-

FINE_FUNCTION will give the term entering in the r.h.s (unit consistency : current).

4. **DEFINE FUNCTION** variables available in EM_RANDLES keywords :

*DEFINE_FUNCTIONs: / *Variable names* :	Randles Circuit parameters (r_0, r_{10}, c_{10} *etc*)	RDLTYPE = -1	Internal Short	Exothermic reaction
'time' : Current EM time	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'emdt' : Current EM timestep	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_sep,y_sep,z_sep' : Positive Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_sen,y_sen,z_sen' : Negative Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_ccp,y_ccp,z_ccp' : Positive Current collector coordinates	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_ccn,y_ccn,z_ccn' : Negative Current collector coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'pres' : Local pressure	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'rho' : Local density	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'vmstress' : Local von Mises stress	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'cond' : Local electrical conductivity	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'temp' : Local Temperature	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'tempRand' : Temperature associated to Randles Circuit	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'efstrain' : Local Effective strain	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models

'strainLocX/Y/Z' : Local strain in the X/Y/Z directions	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'soc,soceff' : Local state of charge, effective state of charge	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'current' : Transverse Randles current	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ocv,vc,volt,r0' : open charge voltage, damping voltage, total voltage, r0 resistance.	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'H_ex' : Exothermal heating power integrated over time (=exothermal heating energy) when exothermic reaction keyword is present.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'short' : Short gives the state of the Randles circuit. If short = 0, then the circuit is not shorted, if short = 1, then it is shorted.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ero : Ero gives the state of the erosion. If ero = 1, then the circuit is adjacent to an element which has been eroded, ero = 0 otherwise.	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models
'areaCircuit,areaCell, *areashortGlob'* : local Randles circuit area, Total Randles Cell area, Total Cell Shorted area.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'save1,save2,save3…,save10' : ten local variables that the user can define and that will be saved during the run and associated to each local Randles circuits.	No	Yes	No	No

*EM_RANDLES_SHORT

Purpose: For battery cell internal short, define conditions to turn on a Randles short (replace one or several Randles circuits by resistances), and to define the value of the short resistance.

Card 1	1	2	3	4	5	6	7	8
Variable	AREATYPE	FUNCTID						
Type	I	I						
Default	none	None						

VARIABLE	DESCRIPTION
AREATYPE	Works the same way as RDLAREA in *EM_RANDLES_SOLID or in *EM_RANDLES_TSHELL : EQ.1: The resistance inverse in FUNCTID is per unit area so that, for each local Randles circuit, the result returned by FUNCTID is multiplied by a factor 1./areaLoc (areaLoc is the local area associated to each Randles circuit while areaGlob is the area of the whole cell). Unit consistency in S.I: Ohms times square meters. EQ.2: Default. The resistance in FUNCTID is for the whole cell (the whole cell is shorted), so that, for each Randles circuit, the result returned by FUNCTID is multiplied by a factor areaGlob/areaLoc . Unit consistency in S.I: Ohms. EQ.3: The resistance returned by FUNCTID is taken as is for each Randles circuit. Unit consistency in S.I : Ohms.
FUNCTID	DEFINE_FUNCTION ID giving the local resistance function of local parameters for the local Randles circuit. See Remark 2.

Remarks:

1. If the return value of the function is zero, there is no short, the Randles circuit is maintained. A positive returned value will replace the Randles circuit by the returned short resistance. In order to ensure that the short is maintained even after the original criteria is no longer met, the default positive value may be

replaced by a negative value. The solver will then take the absolute value returned and adopt it as the new short resistance in case the original short criteria is no longer met rather than reverting to a Randles circuit.

2. The parameter description is :

*DEFINE_FUNCTIONs: / *Variable names* :	Randles Circuit parameters (r_0, r_{10}, c_{10} *etc*)	RDLTYPE = -1	Internal Short	Exothermic reaction
'time' : Current EM time	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'emdt' : Current EM timestep	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_sep,y_sep,z_sep' : Positive Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_sen,y_sen,z_sen' : Negative Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_ccp,y_ccp,z_ccp' : Positive Current collector coordinates	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_ccn,y_ccn,z_ccn' : Negative Current collector coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'pres' : Local pressure	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'rho' : Local density	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'vmstress' : Local von Mises stress	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'cond' : Local electrical conductivity	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'temp' : Local Temperature	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'tempRand' : Temperature associated to Randles Circuit	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'efstrain' : Local Effective strain	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'strainLocX/Y/Z' : Local strain in the X/Y/Z directions	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models

'soc,soceff' : Local state of charge, effective state of charge	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'current' : Transverse Randles current	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ocv,vc,volt,r0' : open charge voltage, damping voltage, total voltage, r0 resistance.	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'H_ex' : Exothermal heating power integrated over time (=exothermal heating energy) when exothermic reaction keyword is present.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'short' : Short gives the state of the Randles circuit. If short = 0, then the circuit is not shorted, if short = 1, then it is shorted.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ero : Ero gives the state of the erosion. If ero = 1, then the circuit is adjacent to an element which has been eroded, ero = 0 otherwise.	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models
'areaCircuit,areaCell, *areashortGlob'* : local Randles circuit area, Total Randles Cell area, Total Cell Shorted area.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'save1,save2,save3…,save10' : ten local variables that the user can define and that will be saved during the run and associated to each local Randles circuits.	No	Yes	No	No

3. An example of a function :

```
*DEFINE_FUNCTION

FID (Function Id)

Float resistance_short_randle(

float time,
```

```
 float x_ccp,float y_ccp,float z_ccp,

float x_sep,float y_sep,float z_sep,

float x_sem,float y_sem,float z_sem,

float x_ccm,float y_ccm,float z_ccm)

{ float seThick0;

seThick0 = 1.e-5;

  seThick=(sqrt(x_sep-x_sem)^2+(y_sep-y_sem)^2+(z_sep-
z_sem)^2);

if (seThick >= seThick0) then

return -1.e-3;

else

return 1.e-2;

endif
```

In this example, as long as seThick is smaller than seThick0, no short occurs. Once seThick becomes larger than seThick0, a short occurs and the short resistance is 1.e-2. If during the run, seThick once again becomes smaller than seThick0, the short is maintained and the short resistance becomes 1.e-3. Replacing 1.e-3 by 0. would cause the short to revert to the original Randles circuit.

*EM_RANDLES_SOLID

Purpose: define the distributed Randles circuit parameters for a Randles cell when using a solid mechanical model.

Card 1	1	2	3	4	5	6	7	8
Variable	RDLID	RDLTYPE	RDLAREA	CCPPART	CCNPART	SEPPART	PELPART	NELPART
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	Q	CQ	SOCINIT	SOCTOU				
Type	F	F	F	F				
Default	none	none	none	none				

Card 3.a	1	2	3	4	5	6	7	8
Variable	R0CHA	R0DIS	R10CHA	R10DIS	C10CHA	C10DIS		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Optional Card. This card only needs to be defined for RDLTYPE greater than 1.

Card 3.b	1	2	3	4	5	6	7	8
Variable	R20CHA	R20DIS	C20CHA	C20DIS	R30CHA	R30DIS	C30CHA	C30DIS
Type	F	F	F	F	F	F	F	F
Default	0.	0.	0.	0.	0.	0.	0.	0.

Optional Thermal card.

Card 4	1	2	3	4	5	6	7	8
Variable	TEMP	FRTHER	ROTOTH	DUDT	TEMPU			
Type	F	I	I	F	I			
Default	0.	0	0	None	0			

Optional SOC shift card

Card 5	1	2	3	4	5	6	7	8
Variable	USESOCS	TAU	FLCID					
Type	I	F	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
RDLID	Id of the Randles Cell

VARIABLE	DESCRIPTION
RDLTYPE	Type of Randles Cell: EQ.-1: User defined equivalent circuit model. See Remark 3. EQ.0: 0-order Randles Cell EQ.1: 1-order Randles Cell EQ.2: 2-order Randles Cell EQ.3: 3-order Randles Cell
RDLAREA	Randles Area: EQ.1: the parameters are per unit area and will be scaled in each Randle circuit by a factor depending on the local area of the circuit. Unit consistency in S.I : Ohms times square meters. EQ.2: the parameters are defined for the whole cell and will be scaled in each Randle circuit by a factor depending on the local area of the circuit and the global area of the cell (default). Unit consistency in S.I : Ohms. EQ.3: the parameters are not scaled by area factors. Unit consistency in S.I : Ohms.
CCPPART	Current Collector Positive Part ID
CCNPART	Current Collector Negative Part ID
SEPPART	Separator Part ID
PELPART	Positive Electrode Part ID
NELPART	Negative Electrode Part ID
Q	Cell capacity.
CQ	SOC conversion factor (%/s), known to be equal to 1/36 in S.I units.
SOCINIT	Initial state of charge of the cell.
SOCTOU	Equilibrium voltage (OCV): GE.0.0: constant value LT.0.0: |SOCTOU| is a load curve ID defining equilibrium voltage (OCV) as a function of the state of charge (SOC).

VARIABLE	DESCRIPTION
R0CHA/ R10CHA/ C10CHA	$r_0/r_{10}/c_{10}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R0DIS/ R10DIS/ C10DIS	$r_0/r_{10}/c_{10}$ when the current flows in the discharge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20CHA/ R30CHA/ C20CHA/ C30CHA	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the charge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the circuit parameters can be made function of the SOC and temperature.
R20DIS/ R30DIS/ C20DIS/ C30DIS	$r_{20}/r_{30}/c_{20}/c_{30}$ when the current flows in the discharge direction: GE.0.0: constant value LT.0.0: absolute value is a define function or table ID. See Remark 4 for the accepted define function arguments. For a define table, the parameters can be made function of the SOC and temperature.
TEMP	Constant temperature value used for the Randles circuit parameters in case there is no coupling with the thermal solver (FRTHER = 0)
FRTHER	From Thermal : EQ.0: The temperature used in the Randles circuit parameters is TEMP. EQ.1: The temperature used in the Randle circuit parameter is the temperature from the thermal solver.

VARIABLE	DESCRIPTION
R0TOTH	r_0 to Thermal: EQ.0: The joule heating in the resistance r0 is not added to the thermal solver. EQ.1: The joule heating in the resistance r0 is added to the thermal solver.
DUDT	If negative integer, load curve ID of the reversible heat as a function of SOC.
TEMPU	Temperature Unit : EQ.0: The temperature is in Celsius EQ.1: The Temperature is in Kelvin
USESOCS	Use SOC shift (See Remark 2) : EQ.0: Don't use the added SOCshift EQ.1: Use the added SOCshift
TAU	Damping time in the SOCshift equation (See Remark 1)
FLCID	Load curve giving f(i) where I is the total current in the unit cell

Remarks:

1. **Element Normal orientation.** the solid element normals must all be oriented in the positive current collector to negative current collector direction in order to detect which current collector nodes are connected to one another. Furthermore, any number of layers can be modelled but the meshes of the CCP, anode, separator, cathode, CCN must be continuous and have merged nodes at the boundaries.

2. **Accounting for Diffusion Limitations.** Sometimes, an extra term called SOCshift (or SocS) can be added at high rate discharges to account for diffusion limitations. The SOCshift is added to SOC for the calculation of the OCV $u(\text{SOC} + \text{SOCshift})$ and $r_0(\text{SOC} + \text{SOCshift})$. SOCshift satisfies the following equation:

$$\frac{d(\text{SOCshift})}{dt} + \frac{\text{SOCshift}}{\tau} = \frac{f(i(t))}{\tau}$$

with $\text{SOCshift}(t = 0) = 0$.

3. **User defined ECMs**. Randles circuits are based on the finite element Robin boundary condition, acting similarly to a convection boundary condition for the heat equation. As such, the circuit equation is decomposed into a term that will enter the left-hand-side of the stiffness matrix and a term that will be added to the right-hand-side. Changing the definition of those terms allows the user to replace the Randles circuits by any type of Equivalent Circuit model (ECM). This is the purpose of RDLTYPE = -1 where the term entering the l.h.s is defined by a negative integer referring to a *DEFINE_FUNCTION ID in R0CHA (unit consistency: Resistance) while a negative integer in R0DIS, associated to a *DEFINE_FUNCTION will give the term entering in the r.h.s (unit consistency : current).

4. **DEFINE FUNCTION** variables available in EM_RANDLES keywords :

*DEFINE_FUNCTIONs: / *Variable names* :	Randles Circuit parameters (r_0, r_{10}, c_{10} *etc*)	RDLTYPE = -1	Internal Short	Exothermic reaction
'time' : Current EM time	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'emdt' : Current EM timestep	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_sep,y_sep,z_sep' : Positive Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_sen,y_sen,z_sen' : Negative Electrode coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'x_ccp,y_ccp,z_ccp' : Positive Current collector coordinates	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'x_ccn,y_ccn,z_ccn' : Negative Current collector coordinates	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models	Solid/Tshell models
'pres' : Local pressure	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'rho' : Local density	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'vmstress' : Local von Mises stress	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'cond' : Local electrical conductivity	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models

'temp' : Local Temperature	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'tempRand' : Temperature associated to Randles Circuit	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'efstrain' : Local Effective strain	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'strainLocX/Y/Z' : Local strain in the X/Y/Z directions	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid /Batmac models	Solid /Batmac models
'soc,soceff' : Local state of charge, effective state of charge	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'current' : Transverse Randles current	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ocv,vc,volt,r0' : open charge voltage, damping voltage, total voltage, r0 resistance.	All models	All models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'H_ex' : Exothermal heating power integrated over time (=exothermal heating energy) when exothermic reaction keyword is present.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'short' : Short gives the state of the Randles circuit. If short = 0, then the circuit is not shorted, if short = 1, then it is shorted.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'ero : Ero gives the state of the erosion. If ero = 1, then the circuit is adjacent to an element which has been eroded, ero = 0 otherwise.	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models	Solid/Batmac models
'areaCircuit,areaCell, *areashortGlob'* : local Randles circuit area, Total Randles Cell area, Total Cell Shorted area.	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models	Solid/Tshell/Batmac models
'save1,save2,save3…,save10' : ten local variables that the user can define and that will be saved during the run and associated to each local Randles circuits.	No	Yes	No	No

*EM_ROTATION_AXIS

Purpose: Define a rotation axis for the EM solver. This is used with the 2D axisymmetric solver. The axis is defined by a point and a direction.

Card 1	1	2	3	4	5	6	7	8
Variable	XP	YP	ZP	XD	YD	ZD	NUMSEC	
Type	F	F	F	F	F	F	I	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
XP, YP, ZP	x, y, and z coordinates of the point
XD, YD, ZD	x, y, and z components of direction of the axis
NUMSEC	Number of Sectors. This field gives the ratio of the full circle to the angular extension of the mesh. This has to be a power of two. For example, NUMSEC = 4 means that the mesh of the part represents one fourth of the total circle. If NUMSEC = 0 for *EM_2DAXI, the solver will replace it with this value.

*EM_SOLVER_BEM

Purpose: Define the type of linear solver and pre-conditioner as well as tolerance for the EM_BEM solve.

Card 1	1	2	3	4	5	6	7	8
Variable	RELTOL	MAXITE	STYPE	PRECON	USELAST	NCYLBEM		
Type	I	I	I	I	I	I		
Default	10^{-6}	1000	2	2	1	5000		

VARIABLE	DESCRIPTION
RELTOL	Relative tolerance for the iterative solvers (PCG or GMRES). If the results are not accurate enough, the user should try to decrease this tolerance. More iterations will then be needed.
MAXITER	Maximum number of iterations
STYPE	Solver type: EQ.1: direct solve – the matrices will then be considered as dense. EQ.2: pre-conditioned gradient method (PCG) - this method allows for block matrices with low rank blocks, and thus reduces memory used. EQ.3: GMRES method - this method allows for block matrices with low rank blocks and thus reduces memory used. The GMRES option only works in serial for now.
PRECON	Preconditioner type for PCG or GMRES iterative solves: EQ.0: no preconditioner EQ.1: diagonal line EQ.2: diagonal block EQ.3: broad diagonal including all neighbor faces EQ.4: LLT factorization.

VARIABLE	DESCRIPTION
USELAST	This is used only for iterative solvers (PCG or GMRES). EQ.-1: Start from 0 as initial guess for solution of the linear system. EQ.1: Starts from the previous solution normalized by the RHS change.
NCYLBEM	Number of electromagnetism cycles between the recalculation of BEM matrices. If a negative value is entered, then the absolute value will refer to a load curve giving NCYCLBEM as a function of time.

Remarks:

1. **USELAST.** Using USELAST = 1 can save many iterations in the subsequent solves if the vector solution of the present solve is assumed to be nearly parallel to the vector solution of the previous solve, as usually happens in time-domain eddy-current problems.

2. **Moving Conductors.** Since the BEM matrices depend on (and only on) the surface node coordinates of the conductors, it is important to recalculate them when the conductors are moving. The frequency with which they are updated is controlled by NCYLBEM. Note that very small values, for example NCYLBEM = 1, should, generally, be avoided since this calculation involves a high computational cost. However, when two conductors are moving and in contact with each other it is recommended to recalculate the matrices at *every* time step.

*EM_SOLVER_BEMMAT

Purpose: Define the type of BEM matrices as well as the way they are assembled.

Card 1	1	2	3	4	5	6	7	8
Variable	MATID							RELTOL
Type	I							F
Default	none							10^{-6}

VARIABLE	DESCRIPTION
MATID	Defines which BEM matrix the card refers to: EQ.1: **P** matrix EQ.2: **Q** matrix EQ.3: **W** matrix
RELTOL	Relative tolerance on the sub-blocks of the matrix when doing low rank approximations. The user should try to decrease these tolerances if the results are not accurate enough. More memory will then be needed.

Remarks:

1. The **W** matrix only exists when the monolithic solver is activated (see *EM_SOLVER_FEMBEM_MONOLITHIC).

*EM_SOLVER_FEM

Purpose: Define some parameters for the EM FEM solver.

Card 1	1	2	3	4	5	6	7	8
Variable	RELTOL	MAXITE	STYPE	PRECON	USELAST	NCYCLFEM		
Type	I	I	I	I	I	I		
Default	10^{-3}	1000	1	1	1	5000		

VARIABLE	DESCRIPTION
RELTOL	Relative tolerance for the iterative solvers (PCG or GMRES). If the results are not accurate enough, you should try to decrease this tolerance. More iterations will then be needed.
MAXITER	Maximum number of iterations
STYPE	Solver type: EQ.1: direct solve EQ.2: Conditioned Gradient Method (PCG)
PRECON	Preconditioner type for PCG. EQ.0: no preconditioner EQ.1: diagonal line
USELAST	This is used only for iterative solvers (PCG). EQ.-1: starts from 0 as initial solution of the linear system. EQ.1: starts from previous solution normalized by the right-hand-side change.
NCYCLFEM	Number of electromagnetism cycles between the recalculation of FEM matrices. If a negative value is entered, then the absolute value will refer to a load curve giving NCYCLFEM function of time.

Remarks:

1. **Starting from Previous Solution.** Using USELAST = 1 can save many iterations in the subsequent solves if the vector solution of the present solve is assumed to be nearly parallel to the vector solution of the previous solve, as usually happens in time-domain eddy-current problems.

2. **Default Values.** The default values are only valid when the PCG resolution method (STYPE = 2) is used. For the default direct solve (STYPE = 1), those values are ignored.

3. **NCYCLFEM.** When the conductor parts are deforming or undergoing changes in their EM material properties (conductivity for example), the FEM matrices should be recalculated more often, so NCYCLFEM may need to be changed.

*EM_SOLVER_FEMBEM

Purpose: Define some parameters for the standard coupling between the EM_FEM and EM_BEM solvers.

Card 1	1	2	3	4	5	6	7	8
Variable	RELTOL	MAXITE	FORCON					
Type	F	I	I					
Default	10^{-2}	50	0					

VARIABLE	DESCRIPTION
RELTOL	Relative tolerance for the FEM/BEM system solve. If the results are not accurate enough, try decreasing this tolerance. A smaller tolerance will, however, require more iterations.
MAXITER	Maximal number of iterations
FORCON	Force convergence: EQ.0: The code stops with an error if no convergence. EQ.1: The code continues to the next time step even if the RELTOL convergence criteria has not been reached.

Remarks:

This keyword couples the FEM and BEM systems with the Richardson method. At each time step, the solver will iterate between the FEM and the BEM system until reaching convergence (based on the choice of RELTOL and MAXITER). The cost for this solve is low. However, to ensure stability, we recommend imposing a limit on the timestep based on the characteristic diffusion time (See *EM_CONTROL_TIMESTEP). Furthermore, it can be unstable whenever magnetic materials are involved (conductor's permeability different than vacuum permeability). The monolithic solver invoked with *EM_SOLVER_FEMBEM_MONOLITHIC aims to remove those two limitations by solving both the FEM and BEM systems in one single monolithic bloc. For such cases, it is, therefore, the recommended choice (See *EM_SOLVER_FEMBEM_MONOLITHIC).

*EM_SOLVER_FEMBEM_MONOLITHIC

Purpose: Replaces *EM_SOLVER_FEMBEM and turns on the monolithic FEM-BEM solver.

Card 1	1	2	3	4	5	6	7	8
Variable	MTYPE	STYPE	ABSTOL	RELTOL	MAXIT			
Type	I	I	F	F	I			
Default	0	0	10^{-6}	10^{-4}	500			

VARIABLE	DESCRIPTION
MTYPE	Monolithic solver type: EQ.0: Direct symmetric solver.
STYPE	Solver type: EQ.0: MINRES iterative solver EQ.1: GMRES iterative solver
ABSTOL	Absolute tolerance
RELTOL	Relative tolerance
MAXIT	Maximum number of iterations

Remarks:

The monolithic solver aims to overcome the limitations of the classic Richardson iterative coupling between the FEM and BEM systems. The monolithic solver offers better stability for large timesteps and for simulations involving ferromagnetic materials. We recommend this method whenever the *EM_MAT_002 keyword is present.

*ICFD

The keyword *ICFD covers all the different options available in the incompressible fluid solver. The keyword cards in this section are defined in alphabetical order:

*ICFD_BOUNDARY_CONJ_HEAT

*ICFD_BOUNDARY_CONVECTION_TEMP

*ICFD_BOUNDARY_FLUX_TEMP

*ICFD_BOUNDARY_FREESLIP

*ICFD_BOUNDARY_FSI

*ICFD_BOUNDARY_FSI_EXCLUDE

*ICFD_BOUNDARY_FSI_FIXED

*ICFD_BOUNDARY_FSWAVE

*ICFD_BOUNDARY_GROUND

*ICFD_BOUNDARY_NAVIERSLIP

*ICFD_BOUNDARY_NONSLIP

*ICFD_BOUNDARY_PERIODIC

*ICFD_BOUNDARY_PRESCRIBED_LEVELSET

*ICFD_BOUNDARY_PRESCRIBED_MOVEMESH

*ICFD_BOUNDARY_PRESCRIBED_PRE

*ICFD_BOUNDARY_PRESCRIBED_SPTRANSP_CONC

*ICFD_BOUNDARY_PRESCRIBED_TEMP

*ICFD_BOUNDARY_PRESCRIBED_TURBULENCE

*ICFD_BOUNDARY_PRESCRIBED_VEL

*ICFD_BOUNDARY_WEAKVEL

*ICFD_BOUNDARY_WINDKESSEL

*ICFD_CONTROL_ADAPT

*ICFD_CONTROL_ADAPT_SIZE

*ICFD_CONTROL_BACKFLOW

*ICFD_CONTROL_CONJ

*ICFD_CONTROL_DEM_COUPLING

*ICFD_CONTROL_EMBEDSHELL

*ICFD_CONTROL_FSI

*ICFD_CONTROL_GAP

*ICFD_CONTROL_GENERAL

*ICFD_CONTROL_IMPOSED_MOVE

*ICFD_CONTROL_LEVELSET

*ICFD_CONTROL_LOAD

*ICFD_CONTROL_MESH

*ICFD_CONTROL_MESH_MOV

*ICFD_CONTROL_MONOLITHIC

*ICFD_CONTROL_OUTPUT

*ICFD_CONTROL_OUTPUT_SUBDOM

*ICFD_CONTROL_OUTPUT_VAR

*ICFD_CONTROL_PARTITION

*ICFD_CONTROL_POROUS

*ICFD_CONTROL_STEADY

*ICFD_CONTROL_SURFMESH

*ICFD_CONTROL_TAVERAGE

*ICFD_CONTROL_TIME

*ICFD_CONTROL_TRANSIENT

*ICFD_CONTROL_TURB_SYNTHESIS

*ICFD_CONTROL_TURBULENCE

*ICFD_DATABASE_AVERAGE

*ICFD_DATABASE_DRAG

*ICFD_DATABASE_FLUX

*ICFD_DATABASE_FLUXSURF

*ICFD_DATABASE_FORCE_DEM

*ICFD_DATABASE_HTC

*ICFD_DATABASE_NODEAVG

*ICFD_DATABASE_NODOUT

*ICFD_DATABASE_NTEMPOUT

*ICFD_DATABASE_POINTAVG

*ICFD_DATABASE_POINTOUT

*ICFD_DATABASE_RESIDUALS

*ICFD_DATABASE_SSOUT

*ICFD_DATABASE_SSOUT_EXCLUDE

*ICFD_DATABASE_TEMP

*ICFD_DATABASE_TIMESTEP

*ICFD_DATABASE_UINDEX

*ICFD_DATABASE_WETNESS

*ICFD_DEFINE_HEATSOURCE

*ICFD_DEFINE_NONINERTIAL

*ICFD_DEFINE_POINT

*ICFD_DEFINE_RESIDENCETIMESOURCE

*ICFD_DEFINE_SOURCE

*ICFD_DEFINE_SPTRANSPSOURCE

*ICFD_DEFINE_TURBSOURCE

*ICFD_DEFINE_WAVE_DAMPING

*ICFD_INITIAL

*ICFD_INITIAL_LEVELSET

*ICFD_INITIAL_SPTRANSP

*ICFD_INITIAL_TEMPNODE

*ICFD_INITIAL_TURBULENCE

*ICFD_MAT

*ICFD_MODEL_NONNEWT

*ICFD_MODEL_POROUS

*ICFD_MODEL_SPECIES_TRANSPORT

*ICFD_PART

*ICFD_PART_VOL

*ICFD_SECTION

*ICFD_SET_NODE

*ICFD_SOLVER_SPLIT

*ICFD_SOLVER_TOL_FSI

*ICFD_SOLVER_TOL_LSET

*ICFD_SOLVER_TOL_MMOV

*ICFD_SOLVER_TOL_MOM

*ICFD_SOLVER_TOL_MONOLITHIC

*ICFD_SOLVER_TOL_PRE

*ICFD_SOLVER_TOL_TEMP

*ICFD_BOUNDARY_CONJ_HEAT

Purpose: Specify which boundary of the fluid domain will exchange heat with the solid.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	CTYPE	VAL	SFLCID				
Type	I	I	F	I				
Default	none	0	0.	0				

VARIABLE	DESCRIPTION
PID	PID of the fluid surface in contact with the solid
CTYPE	Contact type: EQ.0: Constraint approach EQ.1: Mortar contact
VAL	Optional temperature drop if CTYPE = 0 or interface heat transfer coefficient if CTYPE = 1 (high value by default to ensure perfect contact).
SFLCID	Load curve ID used to describe scale factor on VAL value as a function of time; see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_FUNCTION. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time).

*ICFD_BOUNDARY_CONVECTION_TEMP

Purpose: Impose a heat transfer coefficient on the boundary expressed as $h = \frac{q}{T_s - T_b}$

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	HLCID	HSF	TBLCID	TBSF			
Type	I	I	F	I	F			
Default	none	none	1.	none	1.0			

VARIABLE	DESCRIPTION
PID	PID for a fluid surface.
HLCID	Load curve ID to describe the heat transfer coefficient value versus time, see *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
HSF	Load curve scale factor applied on the heat transfer coefficient value. (default = 1.0)
TBLCID	Load curve ID to describe the environment (i.e bulk) temperature value versus time, see *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
TBSF	Load curve scale factor applied on the environment value. (default = 1.0)

*ICFD_BOUNDARY_FLUX_TEMP

Purpose: Impose a heat flux on the boundary expressed as $q = -k\nabla T$

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	LCID	SF	DEATH	BIRTH			
Type	I	I	F	F	F			
Default	none	none	1.	1.E+28	0.0			

VARIABLE	DESCRIPTION
PID	PID for a fluid surface.
LCID	Load curve ID to describe the temperature flux value versus time, see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. . If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
SF	Load curve scale factor. (default = 1.0)
DEATH	Time at which the imposed motion/constraint is removed: EQ.0.0: default set to 10e28
BIRTH	Time at which the imposed pressure is activated starting from the initial abscissa value of the curve

*ICFD_BOUNDARY_FREESLIP

Purpose: Specify the fluid boundary with free-slip boundary condition.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	PID of the fluid surface where a free-slip boundary condition is applied.

*ICFD_BOUNDARY_FSI

Purpose: This keyword defines which fluid surfaces will be considered in contact with the solid surfaces for fluid-structure interaction (FSI) analysis. This keyword should not be defined if *ICFD_CONTROL_FSI is not defined.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	PID of the fluid surface in contact with the solid domain.

*ICFD_BOUNDARY_FSI_EXCLUDE

Purpose: Specify which solid part IDs are excluded from the FSI search. No forces coming from the fluid will be transmitted to those parts.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	Part ID of a part from the solid mechanics problem that is to be excluded from the FSI analysis

*ICFD_BOUNDARY_FSI_FIXED

Purpose: Define fixed fluid surfaces that will be considered for contact with the solid surfaces for FSI. This keyword is similar to *ICFD_BOUNDARY_FSI, except the fluid surface cannot move. This restriction allows a solid surface to "slide" over the fluid and for the exchange of data, such as temperature in CHT applications. This keyword should not be defined if *ICFD_CONTROL_FSI is not defined.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	Part ID of the fluid surface that can be considered for contact with the solid domain

*ICFD_BOUNDARY_FSWAVE

Purpose: Impose a wave inflow boundary condition.

Card Summary:

Card Sets. Include as many sets of the following cards as needed. This input ends with the next keyword ("*") card.

Card 1. This card is required.

PID	WTYPE	H0	WAMP	WLENG	WMAX	SFLCID	WANG

Card 2. This card is included if WTYPE = 7.

WPEAK							

Data Card Definitions:

Include as many of this card and/or sets of this card with the next as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	WTYPE	H0	WAMP	WLENG	WMAX	SFLCID	WANG
Type	I	I	F	F	F	F	I	F
Default	none	none	none	none	none	none	↓	none

VARIABLE	DESCRIPTION
PID	PID for a fluid surface
WTYPE	Wave type: EQ.1: Stokes wave of first order EQ.2: Stokes wave of second order EQ.3: Stokes wave of fifth order EQ.4: Solitary wave EQ.5: Irregular waves using JONSWAP spectrum

VARIABLE	DESCRIPTION
	EQ.6: Irregular waves using One Parameter Pierson-Moskowitz spectrum EQ.7: Irregular waves using Two Parameter Pierson-Moskowitz spectrum
H0	Water level (from the bottom of the channel) for the unperturbed condition
WAMP	Wave amplitude or height for WTYPE = 1 and 4. Significant wave height for WTYPE = 5, 6, and 7.
WLENG	WTYPE.LE.2: Wave length WTYPE.EQ.3: Wave period WTYPE.EQ.4: Not used WTYPE.GE.5: Minimum wave frequency in spectrum (rad/sec)
WMAX	Maximum wave frequency in spectrum (rad/sec) for WTYPE = 5, 6, and 7. Angle between the boundary and the incident waves (in degrees) for WTYPE = 3.
SFLCID	Scale factor LCID on the wave amplitude for WTYPE = 1, 2 and 3. Number of wave modes (default = 1024) for WTYPE = 5, 6, and 7.
WANG	Angle between incoming wave direction and x-axis for z- and y-aligned gravity vector, or angle between incoming wave direction and y-axis for x-aligned gravity vector.

Card included for WTYPE = 7 only

Card 2	1	2	3	4	5	6	7	8
Variable	WPEAK							
Type	F							
Default	none							

VARIABLE	DESCRIPTION
WPEAK	Peak wave frequency in spectrum [rad/sec] for WTYPE = 7.

Remarks:

1. **Peak Wave Frequency for WTYPE = 6.** For the irregular waves using the One Parameter Pierson-Moskowitz spectrum, the peak wave frequency in the spectrum LS-DYNA calculates the peak wave frequency with:

$$0.4\sqrt{\frac{g}{H_s}}$$

Here g the gravity and H_s is the significant wave height input with WAMP.

*ICFD_BOUNDARY_GROUND

Purpose: Specify the fluid boundary with a ground boundary condition. The ground boundary condition is similar to the nonslip boundary condition except that it will keep V = 0 in all circumstances, even if the surface nodes are moving. This is useful in cases where the nodes are allowed to move or translate (using ICFD_BOUNDARY_PRESCRIBED_MOVEMESH for example) but those displacements are only to accommodate for mesh movement and do not correspond to a physical motion.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	PID of the fluid surface where a ground boundary condition is applied.

*ICFD_BOUNDARY_NAVIERSLIP

Purpose: Specify the fluid boundary with the Navier slip boundary condition.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	FRIC						
Type	I	F						
Default	none	0.						

VARIABLE	DESCRIPTION
PID	PID of the fluid surface where a Navier boundary condition is applied
FRIC	Friction coefficient. If a negative value is entered, it will refer to a load curve ID used to describe the friction coefficient value as a function of time; see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_FUNCTION. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time).

Remarks:

1. **About the Navier condition.** It is similar to the regular free slip condition, except a local source term is added based on the choice of the friction coefficient. Since this extra shear is a function of the fluid's velocity, a smaller time step may be needed to reduce the effects of the introduced nonlinearity.

*ICFD_BOUNDARY_NONSLIP

Purpose: Specify the fluid boundary with a non-slip boundary condition.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	PID of the fluid surface where a non-slip boundary condition is applied.

*ICFD_BOUNDARY_PERIODIC

Purpose: Impose various kinds of constraints between two fluid surfaces.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	PTYPE	PID2	PDLCID	AXE	PTID	ANGLE	
Type	I	I	I	I	I	I	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
PID	Part ID for a fluid surface
PTYPE	Boundary type: EQ.1: Periodic rotation boundary condition EQ.2: Periodic reflective boundary condition EQ.3: Sliding mesh boundary condition
PID2	Part ID for the second surface mesh. The boundary condition selected with PTYPE will be applied between PID and PID2. See Remark 1.
PDLCID	Optional load curve ID to describe the pressure drop value between PID and PID2 as a function of time. This curve can be specified with *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION, or *DEFINE_FUNCTION. For *DEFINE_FUNCTION, the following parameters are allowed: f(x,y,z,vx,vy,temp,pres,time).
AXE	The meaning of AXE depends on PTYPE. It only applies for PTYPE = 1 and 3. For the periodic rotation boundary condition (PTYPE = 1): EQ.1: Rotation around *X*-axis EQ.2: Rotation around *Y*-axis EQ.3: Rotation around *Z*-axis For the sliding mesh boundary condition (PTYPE = 3):

VARIABLE	DESCRIPTION
	EQ.0: The contact distance between two faces of PID and PID2 is based on the characteristic local element size. EQ.1: The contact distance between two faces of PID and PID2 is based on the characteristic local element size scaled by a factor given by ANGLE. EQ.2: The contact distance between two faces of PID and PID2 is based on the length given by ANGLE.
PTID	Origin point ID for PTYPE = 1 and PTYPE = 2. See *ICFD_DEFINE_POINT.
ANGLE	Rotation angle for PTYPE = 1. Characterizes contact distance for PTYPE = 3 and AXE ≠ 0.

Remarks:

1. **Selection of PID and PID2.** When the two meshes are of different densities, we recommend selecting the finer mesh to be PID and the coarser mesh to be PID2.

*ICFD_BOUNDARY_PRESCRIBED_LEVELSET

Purpose: Prescribe the fluid height on a boundary.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	PTID	AXE					
Type	I	I	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
PID	PID of the fluid surface where a fluid height will be imposed.
PTID	Point ID specifying the origin of the fluid surface. See *ICFD_DEFINE_POINT.
AXE	Global axis specifying the direction of the fluid: EQ.1: X-axis EQ.2: Y-axis EQ.3: Z-axis

*ICFD_BOUNDARY_PRESCRIBED_MOVEMESH

Purpose: Allows the node of a fluid surface to translate in certain directions using an ALE approach. This is useful in piston type applications or can also be used in certain cases to avoid big mesh deformation.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	dofx	dofy	dofz				
Type	I	I	I	I				
Default	none	1	1	1				

VARIABLE	DESCRIPTION
PID	PID for a fluid surface.
dofx, dofy, dofz	Degrees of freedom in the X,Y and Z directions : EQ.0: degree of freedom left free (Surface nodes can translate in the chosen direction) EQ.1: prescribed degree of freedom (Surface nodes are blocked)

*ICFD_BOUNDARY_PRESCRIBED_PRE

Purpose: Impose a fluid pressure on the boundary.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	LCID	SF	DEATH	BIRTH			
Type	I	I	F	F	F			
Default	none	none	1.	1.E+28	0.0			

VARIABLE	DESCRIPTION
PID	PID for a fluid surface.
LCID	Load curve ID to describe the pressure value versus time, see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. . If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
SF	Load curve scale factor. (default = 1.0)
DEATH	Time at which the imposed motion/constraint is removed: EQ.0.0: default set to 10E28
BIRTH	Time at which the imposed pressure is activated starting from the initial abscissa value of the curve

*ICFD_BOUNDARY_PRESCRIBED_SPTRANSP_CONC

Purpose: Specify the concentration of the transported species at the boundaries.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	LCID						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
PID	Part ID of the boundary with the concentration
LCID	Load curve ID for the curve giving the concentration at the boundary as a function of time

*ICFD_BOUNDARY_PRESCRIBED_TEMP

Purpose: Impose a fluid temperature on the boundary.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	LCID	SF	DEATH	BIRTH			
Type	I	I	F	F	F			
Default	none	none	1.	1.E+28	0.0			

VARIABLE	DESCRIPTION
PID	PID for a fluid surface.
LCID	Load curve ID to describe the temperature value versus time; see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION or *DEFINE_-FUNCTION. . If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x,y,z,vx,vy,vz,temp,pres,time)$.
SF	Load curve scale factor. (default = 1.0)
DEATH	Time at which the imposed temperature is removed: EQ.0.0: default set to 10E28
BIRTH	Time at which the imposed temperature is activated starting from the initial abscissa value of the curve

*ICFD_BOUNDARY_PRESCRIBED_TURBULENCE

Purpose: Optional keyword for strongly imposing turbulence quantities when you select a RANS turbulence model. See *ICFD_CONTROL_TURBULENCE. This keyword is intended for modifying the default boundary conditions at the inlet.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	VTYPE	IMP	LCID	KS	CS		
Type	I	I	I	I	F	F		
Default	none	none	0	none	0.	0.		

VARIABLE	DESCRIPTION
PID	PID for a fluid surface
VTYPE	Variable type: EQ.1: Turbulence kinetic energy (see Remark 1) EQ.2: Turbulence dissipation rate (see Remark 2) EQ.3: Specific dissipation rate (see Remark 3) EQ.4: Modified turbulence viscosity (see Remark 4)
IMP	Imposition method: EQ.0: Direct imposition through value specified by LCID EQ.1: Using turbulence intensity specified by LCID if VTYPE = 1 (see Remark 1). Using turbulence length scale specified by LCID if VTYPE = 2, 3, or 4 (see Remarks 2, 3, and 4). EQ.2: Using turbulence viscosity ratio specified by LCID. Only available for VTYPE = 2 and 3. See Remarks 2 and 3.
LCID	Load curve ID to describe the variable value as a function of time; see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time, k, e, mut).

VARIABLE	DESCRIPTION
KS/CS	Roughness physical height and roughness constant. When defined, the global values of *ICFD_CONTROL_TURBULENCE are replaced for this surface part.

Remarks:

1. **Turbulence Kinetic Energy.** At the inlet, the relationship between the turbulence kinetic energy, k, and the turbulence intensity, I, is given by:

$$k = \frac{3}{2}\left(U_{\text{avg}}^2 I^2\right) \ .$$

   By default, the solver uses an inlet intensity of 0.05 (5%).

2. **Turbulence Dissipation Rate.** At the inlet, if you specify the turbulent dissipation rate using a length scale, l, the following relationship will be used:

$$\epsilon = {C_\mu}^{3/4} \frac{k^{3/2}}{l} \ .$$

   By default, the solver estimates a length scale based on the total height of the channel.

   If you specify the turbulent viscosity ratio, $r = \mu_t/\mu$, the following relationship will be used:

$$\epsilon = \rho C_\mu \frac{k^2}{\mu\, r} \ .$$

3. **Specific Dissipation Rate.** At the inlet, if you specify the specific dissipation rate using a length scale, l, the following relationship will be used:

$$\omega = {C_\mu}^{-1/4} \frac{k^{1/2}}{l} \ .$$

   By default, the solver estimates a length scale based on the total height of the channel.

   If you specify the turbulence viscosity ratio, $r = \mu_t/\mu$, the following relationship will be used:

$$\omega = \rho \frac{k}{\mu\, r} \ .$$

4. **Modified Turbulent Viscosity.** At the inlet, the relationship between the modified turbulent viscosity, $\tilde{\nu}$, and the length scale, l, is given by:

$$\tilde{\nu} = 0.05\sqrt{\frac{3}{2}}\left(U_{\text{avg}}\, l\right) \ .$$

*ICFD_BOUNDARY_PRESCRIBED_VEL

Purpose: Impose the fluid velocity on the boundary.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	DOF	VAD	LCID	SF	VID	DEATH	BIRTH
Type	I	I	I	I	F	I	F	F
Default	none	none	1	none	1.	0	10^{28}	0.0

VARIABLE	DESCRIPTION
PID	PID for a fluid surface.
DOF	Applicable degrees of freedom: EQ.1: x- degree of freedom EQ.2: y- degree of freedom EQ.3: z degree of freedom EQ.4: Normal direction degree of freedom
VAD	Velocity flag: EQ.1: Linear velocity EQ.2: Angular velocity EQ.3: Parabolic velocity profile EQ.4: Activates synthetic turbulent field on part. See *ICFD_CONTROL_TURB_SYNTHESIS.
LCID	Load curve ID used to describe motion value versus time, see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time). For steady state the motion value is a function of the number of iterations instead of time.
SF	Load curve scale factor. (default = 1.0)

VARIABLE	DESCRIPTION
VID	Point ID for angular velocity application point, see *ICFD_DEFINE_POINT.
DEATH	Time at which the imposed motion/constraint is removed: EQ.0.0: default set to 10^{28}
BIRTH	Time at which the imposed motion/constraint is activated starting from the initial abscissa value of the curve

*ICFD_BOUNDARY_WEAKVEL

Purpose: Specify the fluid boundary with a non-slip boundary condition which is imposed in a weak form.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PID	Part ID of the fluid surface where a weak non-slip boundary condition is applied

*ICFD_BOUNDARY_WINDKESSEL

Purpose: This boundary condition imposes the pressure function of circuit parameters where an analogy is made between the pressure and scalar potential as well as between the flux and the current intensity. Such conditions are frequently encountered in hemodynamics.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	WTYPE	R1	C1	R2	L1		
Type	I	I	F	F	F	F		
Default	none	none	0.	0.	0.	0.		

Optional card if WTYPE = 3 or 4.

Card 2	1	2	3	4	5	6	7	8
Variable	P2LCID	C2	R3					
Type	I	F	F					
Default	None	0.	0.					

VARIABLE	DESCRIPTION
PID	PID for a fluid surface
WTYPE	Circuit type (See Remarks) : EQ.1: Windkessel circuit EQ.2: Windkessel circuit with inverted flux EQ.3: CV type circuit EQ.4: CV type circuit with inverted flux
R1/C1/L1/R2/C2	Parameters (Resistances, inductances, capacities) for the different circuits.
P2LCID	Load curve ID describing behavior of P2(t) function of time for CV type circuit.

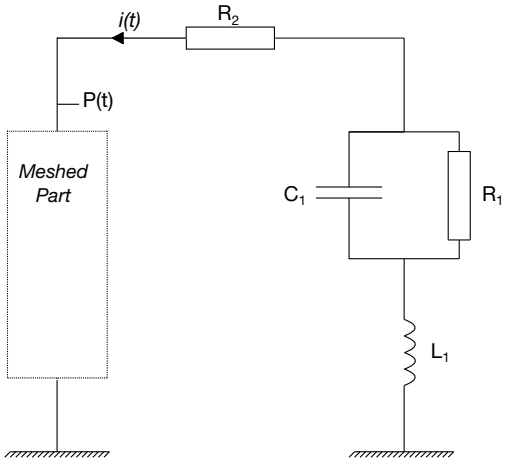


Figure [1]. Windkessel circuit

Remarks:

1. Figure 1 shows a Windkessel circuit and Figure 2 a CV circuit.

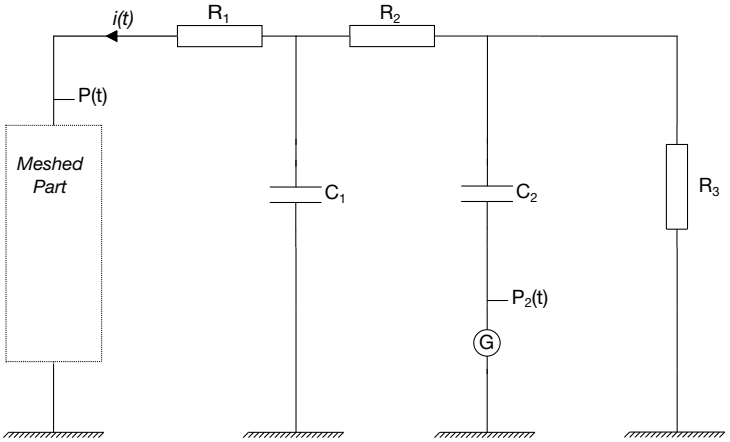


Figure [2]. CV Circuit

*ICFD_CONTROL_ADAPT

Purpose: This keyword will activate the adaptive mesh refinement feature. The solver will use an a-posteriori error estimator to compute a new mesh size bounded by the user to satisfy a maximum perceptual global error.

Card 1	1	2	3	4	5	6	7	8
Variable	MINH	MAXH	ERR	MTH	NIT	VAR		KIS
Type	F	F	F	I	I	I		I
Default	none	none	1.	0	0	0		0

VARIABLE	DESCRIPTION
MINH	Minimum mesh size allowed to the mesh generator. The resulting mesh will not have an element smaller than MINH even if the minimum size does not satisfy the maximum error.
MAXH	Maximum mesh size
ERR	Maximum perceptual error allowed in the whole domain
MTH	Specify if the mesh size is computed based on function error or gradient error: EQ.0: Function error EQ.1: Gradient error
NIT	Number of iterations before a remeshing is forced: GT.0: Number of iterations before a forced remeshing. EQ.0: Do not remesh. LT.0: |NIT| is a load curve ID giving the number iterations before a remeshing as a function of time.
VAR	Specify which variable is taken into account for the error calculation: EQ.0: Velocity, pressure and levelset function are taken into account. EQ.1: Remove the levelset function from the error calculation.

VARIABLE	DESCRIPTION
	EQ.2: Remove the pressure from the error calculation.
	EQ.3: Remove both pressure and levelset function from the error calculation. Only the fluid velocity will, therefore, remain.
KIS	Keep initial mesh size:
	EQ.0: Turned off: The remeshing process will ignore the initial mesh size in the volume.
	EQ.1: Turned on: Whenever a remeshing occurs, the new local mesh size will not be allowed to be substantially coarser than the one from the previous mesh. The object is to diminish the excessive coarsening that can occur between two remeshes.

*ICFD_CONTROL_ADAPT_SIZE

Purpose: This keyword controls the re-meshing of elements taking into account the element quality and distortion in contrast to the default algorithm which only checks for inverted elements.

Card 1	1	2	3	4	5	6	7	8
Variable	ASIZE	NIT	KIS					
Type	I	I	I					
Default	0	none	0					

VARIABLE	DESCRIPTION
ASIZE	EQ.0: only re-mesh in cases where elements invert. EQ.1: re-mesh if elements invert or if element quality deteriorates.
NIT	Number of iterations before a re-meshing is forced. If a negative integer is entered, then a load curve function of time will be used to define NIT.
KIS	Keep initial mesh size: EQ.0: Turned Off: The remeshing process will ignore the initial mesh size in the volume. EQ.1: Turned on: Whenever a remeshing occurs, the new local mesh size will not be allowed to be substantially coarser than the one from the previous mesh. The object is to diminish the excessive coarsening that can occur between two remeshes.

*ICFD_CONTROL_BACKFLOW

Purpose: Modify default values for backflow stabilization.

Card 1	1	2	3	4	5	6	7	8
Variable	BFOR	SF						
Type	I	F						
Default	0	1.0						

VARIABLE	DESCRIPTION
BFOR	Set the backflow stabilization formulation: EQ.0: Default stabilization dependent on spatial velocity gradients. EQ.1: The stabilization adds a temporal velocity gradient which could be necessary for added stabilization.
SF	Scale factor to increase the stabilization if needed. A very small value larger than zero minimizes the effect.

*ICFD_CONTROL_CONJ

Purpose: This keyword allows to pick between the different coupling methods for conjugate heat transfer applications

Card 1	1	2	3	4	5	6	7	8
Variable	CTYPE							TSF
Type	I							F
Default	0							none

VARIABLE	DESCRIPTION
CTYPE	Indicates the thermal coupling type. EQ.0: Robust and accurate monolithic coupling where the temperature field are solved simultaneously between the fluid and the structure. EQ.1: Weak thermal coupling. The fluid passes the heat flux to the solid at the fluid-structure interface and the solid returns the temperature which is applied as a Dirichlet condition.
TSF	Thermal Speedup Factor. This factor multiplies all thermal parameters present in the heat equation with units of time in the denominator (e.g., thermal conductivity, convection heat transfer coefficients). It is used to artificially time scale the thermal problem. A negative value will refer to a time dependent load curve.

Remarks:

1. The keyword ICFD_BOUNDARY_CONJ_HEAT is ignored if CTYPE = 1 but the keyword ICFD_BOUNDARY_FSI is needed in all thermal coupling cases.

*ICFD_CONTROL_DEM_COUPLING

Purpose: Activate coupling between the ICFD and DEM solvers.

Card 1	1	2	3	4	5	6	7	8
Variable	CTYPE	BT	DT	SF	MAXVEL	DTYPE	SFF	FORM
Type	I	F	F	F	F	I	F	I
Default	0	0.	10^{28}	1.	none	0	1.0	0

VARIABLE	DESCRIPTION
CTYPE	Indicates the coupling direction of the solvers: EQ.0: Two-way coupling between the fluid and the solid particles. EQ.1: One-way coupling. The DEM particles transfer their location to the fluid solver. EQ.2: One-way coupling. The fluid solver transfers forces to the DEM particles.
BT	Birth time for the DEM coupling
DT	Death time for the DEM coupling
SF	Scale factor applied to the force transmitted by the fluid to the structure
MAXVEL	Maximal fluid velocity that can be used for the calculation of the fluid force passed on to the DEM particle. This is to avoid having spurious velocities in the fluid causing very high and unrealistic forces on the DEM particles which may lead to a crash.
DTYPE	Drag calculation type: EQ.0: Constant C_d value 0.5 scaled by SF EQ.1: Formula for C_d calculation from Cheng 2009 based on the local Reynolds number value scaled by SF. See Remark 1.
SFF	Scale factor applied to the force transmitted by the structure to the fluid

VARIABLE	DESCRIPTION
FORM	Type of formulation used in the coupling: EQ.0: The force at the particle is based on a velocity drag value. EQ.1: The force is computed using the fluid pressure gradient.

Remarks:

1. **Coefficient of Drag by Cheng 2009.** The calculation for C_d is:

$$C_d = \frac{24}{\mathrm{Re}} \times (1 + 0.27 \times \mathrm{Re})^{0.43} + 0.47 \times \left(1 - \exp\left(-0.04 \times \mathrm{Re}^{0.38}\right)\right)$$

See Cheng 2009 for details.

References:

[1] Cheng, N.-S., "Comparison of formulas for drag coefficient and settling velocity of spherical particles," Powder Technology, 189, 395-398 (2009).

*ICFD_CONTROL_EMBEDSHELL

Purpose: This keyword allows the user to control specific options related to the use of the keyword MESH_EMBEDSHELL.

Card 1	1	2	3	4	5	6	7	8
Variable	GTYPE	DIST	TPS					
Type	I	F	I					
Default	0	0.1	0					

VARIABLE	DESCRIPTION
GTYPE	Gap type. Defines the criteria for selecting a distance to build the gap between the embedded nodes and the newly generated : EQ.0: Automatic and based on the surface mesh size multiplied by a scale factor given by DIST. Default method. EQ.1: Specific gap size given by the user and defined by DIST.
DIST	Distance value if GTYPE = 1 or scale factor value if GTYPE = 0.
TPS	Triple Point Seal. Allows to control the fluid escape through triple points EQ.0: Off. EQ.1: On. The triple points of embedded shells in contact to walls or among each other are sealed and no flow goes through them.

*ICFD_CONTROL_FSI

Purpose: This keyword modifies default values for the fluid-structure interaction coupling algorithm.

Card 1	1	2	3	4	5	6	7	8
Variable	OWC	BT	DT	IDC	LDICSF	XPROJ		
Type	I	F	F	F	I	I		
Default	0	0	10^{28}	0.25	0	0		

Card 2	1	2	3	4	5	6	7	8
Variable	NSUB							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
OWC	Indicates the coupling direction to the solver: EQ.0: Two-way coupling. Loads and displacements are transferred across the FSI interface, and the full non-linear problem is solved. It gives weak FSI coupling when coupled to explicit mechanical solver and strong FSI coupling when coupled to implicit mechanical solver. EQ.1: One-way coupling. The solid mechanics solver transfers displacements to the fluid solver. EQ.2: One-way coupling. The fluid solver transfers stresses to the solid mechanics solver. EQ.3: Two-way coupling. It causes weak coupling (no sub-stepping) with the implicit mechanical solver.

VARIABLE	DESCRIPTION
BT	Birth time for the FSI coupling. Before BT the fluid solver will not pass any loads to the structure, but it will receive displacements from the solid mechanics solver.
DT	Death time for the FSI coupling. After DT the fluid solver will not transfer any loads to the solid mechanics solver, but the fluid will continue to deform with the solid.
IDC	Interaction detection coefficient. See Remark 1.
LCIDSF	Optional load curve ID to apply a scaling factor on the forces transferred to the solid: GT.0: Load curve ID for scale factor as a function of iterations LT.0: |LCIDSF| is a load curve ID for scale factor as a function of time.
XPROJ	Projection of the nodes of the CFD domain that are at the FSI interface onto the structural mesh (see Remark 2): EQ.0: No projection EQ.1: Projection
NSUB	Optional limit on the number of FSI fluid subiterations. This avoids the sometimes unneeded excessive number of FSI subiterations when the fluid and very light structures (like parachutes) develop a resonance-like mode inside the FSI subiterations (coupling iterations).

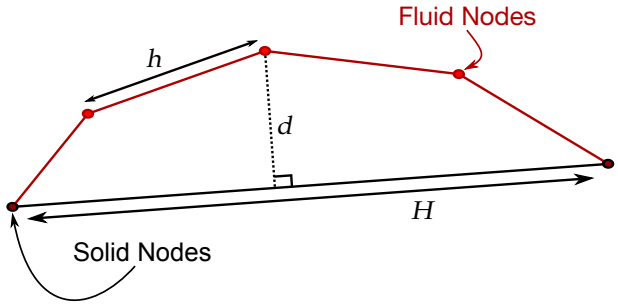


Figure 0-1. Geometry of FSI contact.

Remarks:

1. **Detecting Fluid-Solid Interaction.** One of the criteria to automatically detect the fluid and solid surfaces that will interact in FSI problems is the distance d between a fluid (solid) node and a solid (fluid) element, respectively:

$$d \leq \text{IDC} \times \min(h, H) ,$$

where h is the size of the fluid mesh, H is the size of the solid mechanics mesh, and IDC is a detection coefficient criteria with IDC $= 0.25$ by default. In the majority of cases, this default value is sufficient to ensure FSI interaction. However, it can happen in special cases that the fluid and solid geometries have curvatures that differ too much (such as pipe flows in conjugate heat transfer applications). In such cases, a bigger IDC value may be needed. This flag should be handled with care.

2. **Rotation and Projection of Nodes.** XPROJ = 1 is recommended for cases with rotation.

*ICFD_CONTROL_GAP

Purpose: Activate the gap closure treatment that deals with flow blockage when surfaces come into contact. In ICFD, surface meshes are not allowed to collide or penetrate each other. Rather, activation of the gap closure feature triggers a flow blockage between two surfaces in close proximity based on a user defined contact distance.

Card 1	1	2	3	4	5	6	7	8
Variable	HGAP	MTYPE	DVCL	RDVCL				
Type	F	I	I	I				
Default	none	0	0	0				

Card 2	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
HGAP	Threshold distance. If the distance between surfaces is less than this value, the flow motion is blocked in the gap. Currently there is only one universal value of HGAP for all the surfaces listed in Card 2.
PIDn	Part IDs of the surfaces involved in the gap closure treatment

*ICFD_CONTROL_GENERAL

Purpose: Specify the type of ICFD analysis.

Card 1	1	2	3	4	5	6	7	8
Variable	ATYPE	MTYPE	DVCL	RDVCL	SOLCL			
Type	I	I	I	I	I			
Default	0	0	0	0	0			

VARIABLE	DESCRIPTION
ATYPE	Analysis type: EQ.-1: Turn off the ICFD solver after initial keyword reading EQ.0: Transient analysis (default) EQ.1: Steady state analysis
MTYPE	Solving method type: EQ.0: Fractional Step Method EQ.1: Monolithic solve EQ.2: Potential flow solve
DVCL	Divergence cleaning flag: EQ.0: Initialize the solution with divergence cleaning (default) EQ.1: No divergence cleaning EQ.2: Initial divergence cleaning using potential flow EQ.4: Initial divergence cleaning using steady state solver
RDVCL	Remeshing divergence cleaning: EQ.0: No divergence cleaning after remesh (default) EQ.1: Divergence cleaning after each remeshing step
SOLCL	Solver control: EQ.0: The solver automatically detects if the analysis is 2D or 3D based on element connectivity (default).

VARIABLE	DESCRIPTION
	EQ.1: Turns on the 2D-Axisymmetric solver.

*ICFD_CONTROL_IMPOSED_MOVE

Purpose: Impose a velocity on specific ICFD parts or on the whole volume mesh. Global translation, global rotation, and local rotation components can be defined and combined. This keyword can be used to save calculation time in certain applications, such as sloshing, where the modeling of the whole fluid box and the solving of the consequent FSI problem are not necessarily needed.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	LCVX	LCVY	LCVZ	VADT			
Type	I	I	I	I	I			
Default	none	none	none	none	0			

Optional Card. Rotational velocity components using Euler angles (See Remark 1).

Card 2	1	2	3	4	5	6	7	8
Variable	ALPHAL	BETAL	GAMMAL	ALPHAG	BETAG	GAMMAG	VADR	
Type	I	I	I	I	I	I	I	
Default	0	0	0	0	0	0	0	

Optional Card. Local reference frame definition if ALPHAL, BETAL or GAMMAL used.

Card 3	1	2	3	4	5	6	7	8
Variable	PTID	X1	Y1	Z1	X2	Y2	Z2	
Type	I	F	F	F	F	F	F	
Default	0	1.	0.	0.	0.	1.	0.	

Optional Card. When defined, Cards 2 and 3 are ignored. With this card, rotation is imposed around a point using the velocity of a second point.

Card 4	1	2	3	4	5	6	7	8
Variable	PTID0	AXE	NID					
Type	I	I	I					
Default	0	0	0					

VARIABLE	DESCRIPTION
PID	Part ID. This can be any part ID referenced in *ICFD_PART or *ICFD_PART_VOL. If PID = 0, then the whole volume mesh will be used.
LCVX, LCVY, LCVZ	Load curve IDs for the velocity/displacements in the three global directions (X, Y, Z). To use a *DEFINE_FUNCTION, see Remark 4.
VADT	Velocity/displacements flag for translation components: EQ.0: Prescribe velocity EQ.1: Prescribe displacements
ALPHAL, BETAL, GAMMAL	Load curves IDs for the three Euler angle rotational velocities/displacements in the local reference frame (see Remarks 1 and 2). To use a *DEFINE_FUNCTION, see Remark 4.
ALPHAG, BETAG, GAMMAG	Load curve IDs for the three Euler angle rotational velocities/displacements in the global reference frame (see Remarks 1 and 2). To use a *DEFINE_FUNCTION, see Remark 4.
VADR	Velocity/displacements flag for rotation components: EQ.0: Prescribe velocity EQ.1: Prescribe displacements
PTID	Point ID for the origin of the local reference frame. If not defined, the barycenter of the volume mesh will be used

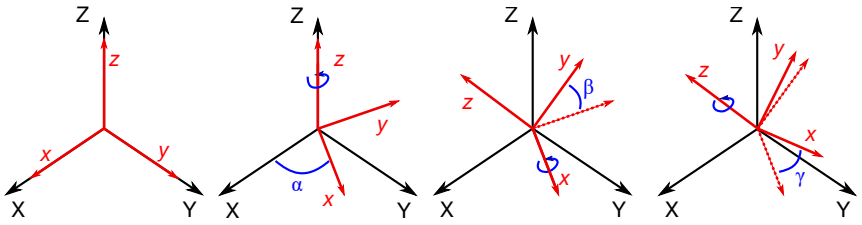


Figure 7-1. A rotation represented by Euler angles (α, β, γ) using $\mathbf{Z}(\alpha)\mathbf{X}(\beta)\mathbf{Z}(\gamma)$ intrinsic rotations.

VARIABLE	DESCRIPTION
X1, Y1, Z1	Three components of the local reference X1 axis. If not defined, the global X axis will be used. See Remark 2.
X2, Y2, Z2	Three components of the local reference X2 axis. If not defined, the global Y axis will be used. See Remark 2.
PTIDO	Point ID (See *ICFD_DEFINE_POINT) for the center of rotation.
AXE	Rotation axis: EQ.1: X-axis EQ.2: Y-axis EQ.3: Z-axis
NID	ICFD surface node ID for the rotational velocity. If the node is static, no rotation will occur. See Remark 3.

Remarks:

1. **Rotations.** Any target orientation can be reached starting from a known reference orientation using a specific sequence of intrinsic rotations whose magnitudes are the Euler angles (α, β, γ). Equivalently, any rotation matrix $\mathbf{R}$ can be decomposed as a product of three elemental rotation matrices. For instance:

$$\mathbf{R} = \mathbf{X}(\alpha)\mathbf{Y}(\beta)\mathbf{Z}(\gamma)$$

However, different definition of the elemental rotation matrices (x, y, z) and their multiplication order can be adopted. The ICFD solver uses the following approach and rotation matrix:

$$\mathbf{Z}(\alpha)\mathbf{X}(\beta)\mathbf{Z}(\gamma) = \begin{bmatrix} c_\alpha c_\gamma - c_\beta s_\alpha s_\gamma & -c_\beta c_\gamma s_\alpha - c_\alpha s_\gamma & s_\alpha s_\beta \\ c_\gamma s_\alpha + c_\alpha c_\beta s_\gamma & c_\alpha c_\beta c_\gamma - s_\alpha s_\gamma & -c_\alpha s_\beta \\ s_\beta s_\gamma & c_\gamma s_\beta & c_\beta \end{bmatrix}$$

where $\boldsymbol{X}(\alpha)$, $\boldsymbol{Y}(\beta)$, and $\boldsymbol{Z}(\gamma)$ *are* the matrices representing the elemental rotations about the axes (x, y, z), $s_\alpha = \sin(\alpha)$, and $c_\beta = \cos(\beta)$.

2. **Local Coordinate Systems.** It is possible to have the ICFD parts or ICFD_-PART_VOLs rotate around the global reference frame but also to define and use a local reference frame by defining its point of origin and two of its vectors $\mathbf{v}_1 = (\text{X1, Y1, Z1})$ and $\mathbf{v}_2 = (\text{X2, Y2, Z2})$ (note that $\mathbf{v}_1$ and $\mathbf{v}_2$ should be orthogonal). The third vector is, then, in the direction of $\mathbf{v}_1 \times \mathbf{v}_2$. See Figure 7-1.

3. **Purpose of NID.** We developed this feature for rotating problems involving FSI and sliding mesh. For example, the airflow can push the blades of a wind turbine and the rotation of the sliding mesh can be prescribed as function of the blade rotation speed.

4. ***DEFINE_FUNCTION.** For each of the load curves, a *DEFINE_FUNCTION can be used. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, visc, pres, time, dt).

*ICFD_CONTROL_LEVELSET

Purpose: This keyword modifies default values for the level set solver.

Card 1	1	2	3	4	5	6	7	8
Variable	LSRST	LSINL	LSMTH					
Type	I	I	F					
Default	20	0	0.0					

VARIABLE	DESCRIPTION
LSRST	This parameter specifies how often the level set distance function is re-initialized. The default value is every 20 time steps.
LSINL	Set the level set to be positive at a velocity inlet: EQ.0: Default is let the level set algorithm compute the value. EQ.1: Force a positive level set value at the inlet.
LSMTH	Scale factor for level set smoothness. GT.0.0: Add smoothness to the free surface. Small values like 0.1 are reasonable. The optimal value could be problem dependent.

*ICFD_CONTROL_LOAD

Purpose: This keyword resets the body load in the ICFD solver to zero, while leaving the body load unchanged for the solid mechanics solver. It is useful in problems where the gravity acceleration may be neglected for the fluid problem, but not for the solid mechanics problem.

Card 1	1	2	3	4	5	6	7	8
Variable	ABL							
Type	I							
Default	1							

VARIABLE	DESCRIPTION
ABL	EQ.0: the body load provided in *LOAD_BODY is reset to zero only for the fluid analysis.

*ICFD_CONTROL_MESH

Purpose: Modify default values for automatic volume mesh generation.

Card 1	1	2	3	4	5	6	7	8
Variable	MGSF		MSTRAT	2DSTRUC	NRMSH			
Type	F		I	I	I			
Default	1.41		0	0	0			

Optional card. This card is optional.

Card 2	1	2	3	4	5	6	7	8
Variable	AVER	SFBL						
Type	I	F						
Default	14	1.0						

VARIABLE	DESCRIPTION
MGSF	Mesh Growth Scale Factor. It specifies the maximum mesh size that the volume mesher is allowed to use when generating the volume mesh based on the mesh surface element sizes defined in *MESH_SURFACE_ELEMENT. See Remark 1.
MSTRAT	Mesh generation strategy (see Remark 2): EQ.0: Mesh generation based on Delaunay criteria EQ.1: Mesh generation based on octree
2DSTRUC	Flag to decide between an unstructured mesh generation strategy in 2D or a structured mesh strategy: EQ.0: Structured mesh EQ.1: Unstructured mesh

VARIABLE	DESCRIPTION
NRMSH	Flag to turn off any remeshing (see Remark 3): EQ.0: Remeshing possible EQ.1: Remeshing not allowed
AVER	Automatic Volume Mesher version (see Remark 4): EQ.14: Version 14 EQ.16: Version 16
SFBL	Scale factor that controls the speed of boundary layer inflation. If the boundary layer is much larger in size than the surface mesh, we recommend a value less than one.

Remarks:

1. **MGSF.** For MGSF, values between 1 and 2 are allowed. Values closer to 1 will result in a finer volume mesh (1 means the volume mesh is not allowed to be coarser than the element size from the closest surface meshes) and values closer to 2 will result in a coarser volume mesh (2 means the volume can use elements as much as twice as coarse as those from the closest surface mesh). MGSF has a fixed value of 1 in 2D.

2. **Mesh Generation Strategy.** The default mesh generation strategy (based on Delaunay criteria) yields a linear interpolation of the mesh size between two surfaces facing each other whereas the octree-based generation strategy allows for the sizes of the elements to remain close to the element surface mesh size over a longer distance. This octree strategy can be useful for creating a smoother transition in configurations where two surface meshes facing each other have very distinct sizes.

3. **NRMSH.** If you know in advance that no remeshing will occur during the analysis, then setting NRMSH to 1 may be useful as it will free up space used to back up the mesh and consequently lower memory consumption.

4. **Version.** Version 14 is the default version used for the ICFD solver automatic volume mesher. Version 16 is now supported and available as option. In some cases, it can yield an approximatively 20% mesh generation speed gain.

*ICFD_CONTROL_MESH_MOV

Purpose: Choose the type of algorithm for mesh movement.

Card 1	1	2	3	4	5	6	7	8
Variable	MMSH	LIM_ITER	RELTOL					
Type	I	I	F					
Default	2	100	10^{-3}					

VARIABLE	DESCRIPTION
MMSH	Mesh motion selector: EQ.-1: Completely shuts off any mesh movement EQ.1: Mesh moves based on the distance to moving walls. EQ.2: Mesh moves by solving a linear elasticity problem using the element sizes as stiffness (default). EQ.3: Mesh uses a Laplacian smoothing with stiffness on edges and from node to opposite faces. Very robust, but costly. EQ.4: Full Lagrangian. The mesh moves with the velocity of the flow. EQ.11: Mesh moves using an implicit ball-vertex spring method. EQ.22: Mesh moves by solving a linear elasticity problem using a constant size. This can be useful to avoid large distortions in rotating problems that involve large discrepancies in mesh sizes (typically in cases involving boundary layer mesh).
LIM_ITER	Maximum number of linear solver iterations for the ball-vertex linear system
RELTOL	Relative tolerance to use as a stopping criterion for the ball-vertex method iterative linear solver (conjugate gradient solver with diagonal scaling preconditioner)

*ICFD_CONTROL_MONOLITHIC

Purpose: This keyword allows to choose between the Fractional Step Solver and the Monolithic Solver.

Card 1	1	2	3	4	5	6	7	8
Variable	SID							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
SID	Solver ID : EQ.0: Fractional Step Solver. Default. EQ.1: Monolithic Solver.

*ICFD_CONTROL_OUTPUT

Purpose: This keyword modifies default values for screen and file outputs related to this fluid solver only.

Card 1	1	2	3	4	5	6	7	8
Variable	MSGL	OUTL	DTOUT	LSPPOUT		ITOUT		
Type	I	I	F	I		I		
Default	0	0	0	0		0		

Optional Card.

Card 2	1	2	3	4	5	6	7	8
Variable	PITOUT							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MSGL	Message level. EQ.0: only time step information is output. EQ.1: first level solver information. EQ.2: full output information with details about linear algebra and convergence steps. EQ.4: full output information is also copied to the messag file.

VARIABLE	DESCRIPTION
OUTL	Output the fluid results in other file formats apart from d3plot. EQ.0: only d3plot output EQ.2: output a file with mesh statistics and the fluid results in OpenDX format. A directory named dx will be created in the work directory where the output files will be written. EQ.6: output a file with mesh statistics and the fluid results in VTK format readable by Paraview. A directory named vtk will be created in the work directory where the output files will be written. EQ.7: output a file with mesh statistic and the fluid results in VTU format readable by Paraview. A directory named vtk will be created in the work directory where the output files will be written. EQ.10: output a file with mesh statistic and the fluid results in Fieldview ASCII format. . A directory named fv will be created in the work directory where the output files will be written. Only available in 3D. EQ.11: output a file with mesh statistic and the fluid results in Fieldview binary format. . A directory named fv will be created in the work directory where the output files will be written. Only available in 3D.
DTOUT	Time interval to print the output when OUTL is different than 0.
LSPPOUT	EQ.0: no LSPP output is produced. EQ.1: outputs a file with the automatically created fluid volume mesh in a format compatible for LSPP at each remesh. Also outputs the fluid volume mesh in a format compatible with a subsequent ICFD analysis. EQ.3: Outputs the fluid volume mesh in a format compatible with a subsequent ICFD analysis at each DTOUT.
ITOUT	Iteration interval to print the output, including the d3plot files when the steady state solver is selected (See ICFD_CONTROL_GENERAL).

VARIABLE	DESCRIPTION
PITOUT	Pressure iteration limit output. If the number of pressure iterations in the fractional step solve goes above PITOUT, an extra d3plot will be dumped. This is mainly a debugging feature which can help the user identify problematic areas in the model which often precede a divergence.

.

*ICFD_CONTROL_OUTPUT_SUBDOM

Purpose: Defines a specific zone that should be output in the format specified by the ICFD_CONTROL_OUTPUT card rather than the whole domain.

Shape Control. First card specifies the shape of the output sub domain.

Card 1	1	2	3	4	5	6	7	8
Variable	SNAME							
Type	A							
Default	none							

Box Case. Card 2 for Sname = box

Cards 2	1	2	3	4	5	6	7	8
Variable	PMINX	PMINY	PMINZ	PMAXX	PMAXY	PMAXZ		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

Sphere Case. Card 2 for Sname = sphere

Cards 3	1	2	3	4	5	6	7	8
Variable	RADIUS	CENTERX	CENTERY	CENTERZ				
Type	F	F	F	F				
Default	none	none	none	none				

Cylinder Case. Card 2 for Sname = cylinder

Cards 4	1	2	3	4	5	6	7	8
Variable	Radius	PMINX	PMINY	PMAXZ	PMAXX	PMAXY	PMAXZ	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
SNAME	Shape name. Possibilities include 'box', 'cylinder' and 'sphere'
PMINX, Y, Z]	X, Y, Z for the point of minimum coordinates
PMAX[X, Y, Z]	X, Y, Z for the point of maximum coordinates
CEN-TER[X, Y, Z]	Coordinates of the sphere center in cases where Sname is Sphere
RADIUS	Radius of the sphere if SNAME is *sphere* or of the cross section disk if SNAME is *cylinder*.

*ICFD_CONTROL_OUTPUT_VAR

Purpose: This keyword allows the user to turn off the output of certain CFD variables to reduce the size of the d3plot files.

Card 1	1	2	3	4	5	6	7	8
Variable	VEL	AVGVEL	VORT					
Type	I	I	I					
Default	0	0	0					

Card 2	1	2	3	4	5	6	7	8
Variable	PRE	PREAVG	LSET	QC	CFL			
Type	I	I	I	I	I			
Default	0	0	0	0	0			

Card 3	1	2	3	4	5	6	7	8
Variable	TEMP	TEMPAVG						
Type	I	I						
Default	0	0						

Card 4	1	2	3	4	5	6	7	8
Variable	KP	EP	MUT	INT	CMU			
Type	I	I	I	I	I			
Default	0	0	0	0	0			

VARIABLE	DESCRIPTION
VEL/AVGVEL/ VORT	Velocity, average velocity, vorticity : EQ.0: Is output. EQ.1: Is not output.
PRE/PREAVG/ LSET/QC/CFL	Pressure, average pressure, levelset, Q criterion, CFL number : EQ.0: Is output. EQ.1: Is not output.
TEMP/ TEMPAVG	Temperature, average temperature : EQ.0: Is output. EQ.1: Is not output.
KP/EP/MUT /INT/CMU	RANS output variables, kinetic energy, diffusion, turbulent viscosity, turbulent intensity, Cmu variable : EQ.0: Is output. EQ.1: Is not output.

.

*ICFD_CONTROL_PARTITION

Purpose: This keyword changes the default option for the partition in MPP, thus it is only valid in MPP.

Card 1	1	2	3	4	5	6	7	8
Variable	PTECH							
Type	I							
Default	1							

VARIABLE	DESCRIPTION
PTECH	Indicates the type of partition: EQ.1: the library Metis is used. EQ.2: partition along the axis with higher aspect ratio EQ.3: partition along *X*-axis EQ.4: partition along *Y*-axis EQ.5: partition along *Z*-axis

*ICFD_CONTROL_POROUS

Purpose: This keyword modifies the porous media solve.

Card 1	1	2	3	4	5	6	7	8
Variable	PMSTYPE	VELMETH						
Type	I	I						
Default	0	0						

VARIABLE	DESCRIPTION
PMSTYPE	Indicates the porous media solve type. EQ.0: Anisotropic Generalized Navier-Stokes model for porous media (See *ICFD_MODEL_POROUS) using Fractional step method. EQ.1: Anisotropic Darcy-Forcheimer model using a Monolithic approach for the solve. This method is better suited for very low Reynolds flows through porous media (frequently encountered in Resin Transfer Molding (RTM) applications). See Remark 1.
VELMETH	Method for determining advection velocity: EQ.0: Uses FEM approximation for advection velocity EQ.1: Uses PFEM2 for advection velocity

Remarks:

1. **Anisotropic Darcy-Forcheimer Model.** When using the Anisotropic Darcy-Forcheimer model, the convective term in the Navier Stokes formulation is neglected.

*ICFD_CONTROL_STEADY

Purpose: This keyword allows to specify convergence options for the steady state solver.

Card 1	1	2	3	4	5	6	7	8
Variable	ITS	TOL1	TOL2	TOL3	REL1	REL2	UREL	ORDER
Type	I	F	F	F	F	F	F	I
Default	1e6	1.e-3	1.e-3	1.e-3	0.3	0.7	1.	0

VARIABLE	DESCRIPTION
ITS	Maximum number of iterations to reach convergence.
TOL1/2/3	Tolerance limits for the momentum pressure and temperature equations respectfully.
REL1/2	Relaxation parameters for the velocity and pressure respectfully. Decreasing those values may add stability but more iterations may be needed to reach convergence.
UREL	Under relaxation parameter. Lowering this value may improve the final accuracy of the solution but more iterations may be needed to achieve convergence.
ORDER	Analysis order : EQ.0: Second order. More accurate but more time consuming. EQ.1: First order: More stable and faster but may be less accurate.

*ICFD_CONTROL_SURFMESH

Purpose: This keyword enables automatic surface re-meshing. The objective of the re-meshing is to improve the mesh quality on the boundaries. It should not be used on a regular basis.

Card 1	1	2	3	4	5	6	7	8
Variable	RSRF	SADAPT						
Type	I	I						
Default	0	0						

VARIABLE	DESCRIPTION
RSRF	Indicates whether or not to perform a surface re-meshing. EQ.0: no re-meshing is applied. EQ.1: Laplacian smoothing surface remeshing EQ.2: Curvature preserving surface remeshing
SADAPT	Indicates whether or not to trigger adaptive surface remeshing. EQ.0: no adaptive surface re-meshing is applied. EQ.1: automatic surface remeshing when quality deteriorates (3D only).

.

*ICFD_CONTROL_TAVERAGE

Purpose: This keyword controls the restarting time for computing the time average values. By default, there is no restarting and the average quantities are given starting from $t = 0$. This keyword can be useful in turbulent problems that admit a steady state.

Card 1	1	2	3	4	5	6	7	8
Variable	DT							
Type	F							
Default	none							

VARIABLE	DESCRIPTION
DT	Over each DT time interval, the average quantities are reset.

*ICFD_CONTROL_TIME

Purpose: Change the default values related to time parameters for the fluid problem.

Card 1	1	2	3	4	5	6	7	8
Variable	TTM	DT	CFL	LCIDSF	DTMIN	DTMAX	DTINIT	TDEATH
Type	F	F	F	I	F	F	F	F
Default	10^{28}	0.0	1.0	0	10^{-9}	10^{28}	↓	10^{28}

Optional card

Card 2	1	2	3	4	5	6	7	8
Variable	DTT							
Type	F							
Default	Rem 1							

Optional card

Card 3	1	2	3	4	5	6	7	8
Variable	BTBL							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
TTM	Total time of simulation for the fluid problem
DT	Time step for the fluid problem. If nonzero, the time step will be constant and equal to this value. If set to 0.0, then the time step is automatically computed based on the CFL condition.

VARIABLE	DESCRIPTION
CFL	CFL number for DT = 0.0. In general, CFL specifies a scale factor that is applied to the time step. When DT = 0.0, the time step is set to the maximum value satisfying the CFL condition, in which case this scale factor is equal to the *CFL number*.
LCIDSF	Load curve ID specifying the CFL number when DT = 0.0 as a function of time, and more generally LCIDSF specifies the time step scale factor as a function of time.
DTMIN	Minimum time step. When an automatic time step is used and DTMIN is defined, the time step cannot drop below DTMIN. A negative value will refer to a time dependent load curve.
DTMAX	Maximum time step. When an automatic time step is used and DTMAX is defined, the time step cannot increase beyond DTMAX. A negative value will refer to a time dependent load curve.
DTINIT	Initial time step. If not defined, the solver will automatically determine an initial time step based on the flow velocity or dimensions of the problem in cases where there is no inflow.
TDEATH	Death time for the Navier Stokes solve. After TDEATH, the velocity and pressure will no longer be updated. But the temperature and other similar quantities still can.
DTT	Thermal time step. See Remark 1.
DTBL	Flag to include boundary layer elements in the automatic time step calculation. EQ.0: The boundary layer elements are excluded (default). EQ.1: The boundary layer elements are included.

Remarks:

1. **Thermal Time Step.** By default, the heat equation is solved using the same time step as that for the velocity/pressure system. This option allows you to assign a specific time step for the thermal solve. It can be useful in cases where the time scales are very different between the two domains. When defined, we recommend DTT to always be greater than or equal to the regular CFD time step.

*ICFD_CONTROL_TRANSIENT

Purpose: This keyword allows to specify different integration scheme options for the transient solver.

Card 1	1	2	3	4	5	6	7	8
Variable	TORD	FSORD						
Type	I	I						
Default	0	0						

VARIABLE	DESCRIPTION
TORD	Time integration order : EQ.0: Second order. EQ.1: First order.
FSORD	Fractional step integration order : EQ.0: Second order. EQ.1: First order.

*ICFD_CONTROL_TURBULENCE

Purpose: Modify the default values for the turbulence model.

Card Summary:

Card 1. This card is required.

TMOD	SUBMOD	WLAW	KS	CS		TWLAW	TYPLUS

Card 2a. This card is read if TMOD = 1. It is optional.

CE1	CE2	SIGMAEPS	SIGMAK	CMU	CCUT		

Card 2b. This card is read if TMOD = 2 or 3. It is optional.

Cs							

Card 2c.1. This card is read if TMOD = 4. It is optional.

GAMMA	BETA01	SIGMAW1	SIGMAK1	BETA0ST	CCUT		

Card 2c.2. This card is read if TMOD = 4. It is optional.

A1	BETA02	SIGMAW2	SIGMAK2	CL			

Card 2d. This card is read if TMOD = 5. It is optional.

CB1	CB2	SIGMANU	CNU1	CW1	CW2		

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	TMOD	SUBMOD	WLAW	KS	CS		TWLAW	TYPLUS
Type	I	I	I	F	F		I	F
Default	0	1	1	0.	0.		none	none

VARIABLE	DESCRIPTION
TMOD	Indicates what turbulence model will be used.

VARIABLE	DESCRIPTION
	EQ.0: Turbulence model based on a variational multiscale approach is used by default. EQ.1: RANS k - ε approach (see Remark 1) EQ.2: LES Smagorinsky or dynamic sub-grid scale model EQ.3: LES Wall adapting local eddy-viscosity (WALE) model EQ.4: RANS k - w approach (see Remark 2) EQ.5: RANS Spalart-Allmaras approach
SUBMOD	Turbulence sub-model. For RANS k - ε approach (TMOD = 1): EQ.1: Standard model EQ.2: Realizable model For LES Smagorinsky or dynamic sub-grid model (TMOD = 2): EQ.1: Smagorinsky model (see Remark 6) EQ.2: Dynamic model (see Remark 7). For RANS k - ω approach (TMOD = 4): EQ.1: Standard Wilcox 98 model. EQ.2: Standard Wilcox 06 model. EQ.3: SST Menter 2003.
WLAW	Law of the wall ID if a RANS turbulence model is selected (see Remark 4): EQ.1: Standard classic law of the wall (default for TMOD = 1) EQ.2: Standard Launder and Spalding law of the wall EQ.4: Nonequilibrium Launder and Spalding law of the wall EQ.5: Automatic classic law of the wall
KS	Roughness physical height, only used for RANS turbulence models.
CS	Roughness constant, only used for RANS turbulence models.
TWLAW	Thermal law of the wall flag (see Remark 8): EQ.0: No thermal law of the wall activated.

VARIABLE	DESCRIPTION
	EQ.1: Thermal law of the wall
TYPLUS	Thermal Y+ value (Y_{+t}). If Y_{+t} is not defined, the solver will automatically estimate its value using $Y_{+tc} = Y_{+c}/\mathrm{Pr}^{1./3.}$ with $Y_{+c} = 11.225$ the critical Y_+ value and Pr the Prandtl number.

RANS $k - \varepsilon$ Card. Optional card if TMOD = 1. Optional card read if TMOD = 1. See Remark 1.

Card 2a	1	2	3	4	5	6	7	8
Variable	CE1	CE2	SIGMAE	SIGMAK	CMU	CCUT		
Type	F	F	F	F	F	F		
Default	1.44	1.92	1.3	1.0	0.09	-1.		

VARIABLE	DESCRIPTION
CEPS1	k - ε model constant, $C_{1\varepsilon}$
CEPS2	k - ε model constant, $C_{2\varepsilon}$
SIGMAEPS	k - ε model constant, σ_ε
SIGMAK	k - ε model constant, σ_k
CMU	k - ε model constant, C_μ
CCUT	k - ε model constant, C_{cut}

LES Card. Optional card read if TMOD = 2 or 3.

Card 2b	1	2	3	4	5	6	7	8
Variable	Cs							
Type	F							
Default	0.18							

VARIABLE	DESCRIPTION
Cs	Smagorinsky constant if TMOD = 2 and SUBMOD = 1 or WALE constant if TMOD = 3

RANS $k-\omega$ Card. Optional card read if TMOD = 4. See Remark 2.

Card 2c.1	1	2	3	4	5	6	7	8
Variable	GAMMA	BETA01	SIGMAW1	SIGMAK1	BETA0ST	CCUT		
Type	F	F	F	F	F	F		
Default	1.44	0.072	2.	2.	0.09	-1.		

RANS $k-\omega$ Card. Optional card read if TMOD = 4. See Remark 2.

Card 2c.2	1	2	3	4	5	6	7	8
Variable	A1	BETA02	SIGMAW2	SIGMAK2	CL			
Type	F	F	F	F	F			
Default	0.31	0.0828	2	2	0.875			

VARIABLE	DESCRIPTION
GAMMA	k -ω model constant, γ
BETA01	k -ω model constant, β_{01}
SIGMAW1	k -ω model constant, $\sigma_{\omega 1}$
SIGMAK1	k -ω model constant, σ_{k1}
BETA0ST	k -ω model constant, β_0^*
CCUT	k -ω model constant, C_{cut}
A1	k -ω model constant, a_1
BETA02	k -ω model constant, β_{02}
SIGMAW2	k -ω model constant, $\sigma_{\omega 2}$

VARIABLE	DESCRIPTION
SIGMAK2	k -ω model constant, σ_{k2}
CL	k -ω model constant, C_l

RANS Spalart-Allmaras Card. Optional card read if TMOD = 5.

Card 2d	1	2	3	4	5	6	7	8
Variable	CB1	CB2	SIGMANU	CNU1	CW1	CW2		
Type	F	F	F	F	F	F		
Default	0.1355	0.622	0.66	7.2	0.3	2.0		

VARIABLE	DESCRIPTION
CB1	Spalart-Allmaras constant, C_{b1}
CB2	Spalart-Allmaras constant, C_{b2}
SIGMANU	Spalart-Allmaras constant, σ_ν
CNU1	Spalart-Allmaras constant, $C_{\nu 1}$
CW1	Spalart-Allmaras constant, C_{w1}
CW2	Spalart-Allmaras constant, C_{w2}

Remarks:

1. **k -ε Model (TMOD = 1).** For the Standard k - ε model, the following two equations are solved for the turbulent kinetic energy (k) and the turbulent dissipation (ε):

$$\frac{\partial k}{\partial t} + \frac{\partial (k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[(\frac{\mu}{\rho} + \frac{\mu_t}{\rho \sigma_k}) \frac{\partial k}{\partial x_j} \right] + P_k + P_b - \varepsilon + S_k$$

$$\frac{\partial \varepsilon}{\partial t} + \frac{\partial (\varepsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[(\frac{\mu}{\rho} + \frac{\mu_t}{\rho \sigma_\varepsilon}) \frac{\partial \varepsilon}{\partial x_j} \right] + C_{1\varepsilon} \frac{\varepsilon}{k} P_k - C_{2\varepsilon} \frac{\varepsilon^2}{k} + S_e$$

Here P_k is the k production term (see Remark 3), P_b is the production term due to buoyancy and S_k and S_e are the user defined source terms. P_k and P_b are expressed as:

$$P_k = \frac{\mu_t}{\rho} S^2$$
$$P_b = \frac{\beta \mu_t}{\rho \mathrm{Pr}_t} g_i \frac{\partial T}{\partial x_i}$$

where S is the modulus of the mean rate of strain tensor ($S^2 = 2S_{ij}S_{ij}$), β is the coefficient of thermal expansion, and Pr_t is the turbulent Prandtl number. The turbulent viscosity is then expressed as:

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon}$$

For the realizable k - ε model, the equation for the turbulent kinetic energy does not change, but the equation for the turbulent dissipation is now expressed as:

$$\frac{\partial \varepsilon}{\partial t} + \frac{\partial (\varepsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[(\frac{\mu}{\rho} + \frac{\mu_t}{\rho\, \sigma_\varepsilon}) \frac{\partial \varepsilon}{\partial x_j} \right] + C_1 S \varepsilon - C_{2\varepsilon} \frac{\varepsilon^2}{k + \sqrt{\frac{\mu}{\rho}\varepsilon}} - \varepsilon + S_e \ .$$

Here $C_1 = \max\left[0.43, \frac{\eta}{\eta+5}\right]$ with $\eta = S\frac{k}{\varepsilon}$.

Furthermore, while the turbulent viscosity is still expressed the same way, C_μ is no longer a constant:

$$C_\mu = \frac{1}{A_0 + A_s k \frac{U^*}{\epsilon}} \ .$$

In the above,

$$U^* = \sqrt{\Omega_{ij}\Omega_{ij} + S_{ij}S_{ij}}$$
$$A_0 = 4.04$$
$$A_s = \sqrt{6} cos\left(\frac{1}{3} cos^{-1}\left(\sqrt{6} \frac{S_{ij}S_{jk}S_{ki}}{(S_{ij}S_{ij})^{3/2}} \right) \right)$$

Note that in this case, the constant value C_μ that can be input by you serves as the limiting value that C_μ can take. By default, $C_\mu = 0.09$ so:

$$0.0009 < C_\mu < 0.09$$

2. ***k* - *ω* Model (TMOD = 4).** For the Standard Wilcox 06 k - ω model, the following two equations are solved for the turbulent kinetic energy and the specific turbulent dissipation rate, respectively k and ω:

$$\frac{\partial k}{\partial t} + \frac{\partial (k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\frac{\mu}{\rho} + \frac{\mu_t}{\rho \sigma_{k1}} \right) \frac{\partial k}{\partial x_j} \right] + P_k - \beta^* k\omega + S_k$$
$$\frac{\partial \omega}{\partial t} + \frac{\partial (\omega u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\frac{\mu}{\rho} + \frac{\mu_t}{\rho \sigma_{\omega 1}} \right) \frac{\partial \varepsilon}{\partial x_j} \right] + \gamma \frac{\omega}{k} P_k - \beta \omega^2 + \sigma_d X_k \omega^2 + S_\omega$$

Here P_k is the k production term (see Remark 3) and S_k and S_ω are the user defined source terms. P_k, β^*, β and σ_d are expressed as:

$$\begin{aligned} P_k &= \frac{\mu_t}{\rho} S^2 \\ \beta^* &= \beta_0^* f_{\beta*} \\ \beta &= \beta_{01} f_\beta \\ \sigma_d &= \begin{cases} 0. & X_k \le 0. \\ 1/8 & X_k > 0. \end{cases} \end{aligned}$$

where

$$\begin{aligned} f_\beta &= \frac{1 + 85 X_\omega}{1 + 100 X_\omega} \\ f_{\beta*} &= 1. \\ X_k &= \frac{1}{\omega^3} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j} \\ X_\omega &= \left| \frac{\Omega_{ij} \Omega_{jk} S_{ki}}{(\beta_0^* \omega)^3} \right| \end{aligned}$$

The turbulent viscosity is then:

$$\mu_t = \rho \frac{k}{max\left[\omega, C_l \sqrt{\frac{2 S_{ij} S_{ij}}{\beta_0^*}}\right]}$$

For the Standard Wilcox 98 model, the following terms are modified:

$$\begin{aligned} f_\beta &= \frac{1 + 70 X_\omega}{1 + 80 X_\omega} \\ f_{\beta*} &= \begin{cases} 1 & \text{if } X_k \le 0. \\ \dfrac{1 + 680\ {X_k}^2}{1 + 400\ {X_k}^2} & \text{if } X_k > 0. \end{cases} \\ \sigma_d &= 0. \end{aligned}$$

The turbulent viscosity is then:

$$\mu_t = \rho \frac{k}{\omega}$$

For the Menter SST 2003 model, the following equations are solved:

$$\frac{\partial k}{\partial t} + \frac{\partial (k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j} \left[(\frac{\mu}{\rho} + \frac{\mu_t}{\rho\ \sigma_k}) \frac{\partial k}{\partial x_j} \right] + P_k - \beta_0^* k \omega + S_k$$

$$\frac{\partial \omega}{\partial t} + \frac{\partial (\omega u_i)}{\partial x_i} = \frac{\partial}{\partial x_j} \left[(\frac{\mu}{\rho} + \frac{\mu_t}{\rho\ \sigma_\omega}) \frac{\partial \varepsilon}{\partial x_j} \right] + \frac{\gamma}{\mu_t} P_k - \beta \omega^2 + 2(1 - F_1)\ \sigma_{\omega 2} X_k \omega^2 + S_\omega$$

Each of the constants, γ, β, σ_k, and σ_ω are now computed by a blend of two constants with a blending function through:

$$\alpha = \alpha_1 F_1 + \alpha_2 (1 - F_1)$$

The blending function F_1 is defined by:

$$F_1 = \tanh \left\langle \left[\min \left(\max \left(\frac{\sqrt{k}}{\beta_0^* \omega y}, \frac{500\nu}{y^2 \omega} \right), \frac{4\rho\sigma_{\omega 2} k}{\mathrm{CD} \times y^2} \right) \right]^4 \right\rangle$$

With y the distance to the nearest wall and:

$$\mathrm{CD} = \max(2\rho\sigma_{\omega 2} X_k \omega^2, 10^{-10})$$

The turbulent viscosity is then:

$$\mu_t = \rho \frac{a_1 k}{\max(a_1 \omega, S F_2)}$$

with:

$$F_2 = \tanh \left[\left(\max \left(\frac{2\sqrt{k}}{\beta_0^* \omega y}, \frac{500\nu}{y^2 \omega} \right) \right)^2 \right]$$

3. **Production Term.** You can activate a limiter on the production term, P_k, for TMOD =1 and 4. If $C_{\mathrm{cut}} \geq 0$ (CCUT), then:

$$P_k = \begin{cases} \min(P_k, C_{\mathrm{cut}} \varepsilon) & \text{if TMOD} = 1 \\ \min(P_k, C_{\mathrm{cut}} \beta_0^* k \omega) & \text{if TMOD} = 4 \end{cases}$$

This is especially common when using the Menter SST 2003 model.

4. **Laws of the Wall for RANS Models.** For RANS models, the following laws of the wall are available:

a) *Standard Classic (WLAW = 1).*

$$U^+ = \begin{cases} \frac{1}{\kappa} \ln(EY^+) & \text{if } Y^+ > 11.225 \\ Y^+ & \text{otherwise} \end{cases}$$

$$Y^+ = \frac{\rho y U_\tau}{\mu}$$

$$U^+ = \frac{U}{U_\tau}$$

$$U_\tau = \sqrt{\frac{\tau_w}{\rho}}$$

This is the default for TMOD = 1.

b) *Standard Launder and Spalding (WLAW = 2).*

$$U^* = \begin{cases} \frac{1}{\kappa} \ln(EY^*) & \text{if } Y^* > 11.225 \\ Y^* & \text{otherwise} \end{cases}$$

$$Y^* = \frac{\rho {C_\mu}^{1/4} k^{1/2} y}{\mu}$$
$$U^* = \frac{U {C_\mu}^{1/4} k^{1/2}}{{U_\tau}^2}$$
$$U_\tau = \sqrt{\frac{\tau_w}{\rho}}$$

c) *Nonequilibrium Launder and Spalding (WLAW = 3).* The nonequilibrium laws of the wall modify the expression of the velocity at the wall making it sensitive to the pressure gradient :

$$\tilde{U} = U - \frac{1}{2}\frac{dp}{dx}\left[\frac{y_v}{\rho\kappa\sqrt{k}} ln\left(\frac{y}{y_v}\right) + \frac{y - y_v}{\rho\kappa\sqrt{k}} + \frac{{y_v}^2}{\mu}\right]$$

with:

$$y_v = \frac{11.225}{y^*} y$$

This law is recommended with TMOD = 1 and in cases of complex flows involving separation, reattachment and recirculation.

d) *Automatic Classic (WLAW = 4).* The automatic wall law attempts to blend the viscous and log layers to better account for the transition zone. In the buffer region, we have :

$$U^+ = \frac{U}{U_\tau}$$
$$U_\tau = \sqrt[4]{(\frac{U}{y^+})^4 + (\frac{U}{\frac{1}{\kappa}\ln\,(Ey^+)})^4}$$

This is the recommended approach for TMOD = 4.

5. **RANS Turbulence Model with Roughness Included.** When a RANS turbulence model is selected, it is possible to define extra parameters to account for roughness effects. In such cases, an extra term will be added to the logarithmic part of the different laws of the wall:

$$U^+ = \frac{1}{\kappa}\ln(E\ Y^+) - \Delta B$$

If we introduce the non-dimensional roughness height:

$$K^+ = \frac{\rho K_s {C_\mu}^{1/4} k^{1/2}}{\mu}\ ,$$

we have:

$$\Delta B = \begin{cases} 0 & \text{for } K^+ \le 2.25 \\ \frac{1}{\kappa}\ln\left(\frac{K^{\pm 2.25}}{87.75} + C_s K^+\right) \times \sin\left(0.4258(\ln K^+ - 0.811)\right) & \text{for } 2.25 < K^+ \le 90.0 \\ \frac{1}{\kappa}\ln(1 + C_s K^+) & \text{for } 90. < K^+ \end{cases}$$

6. **LES Smagorinksy.** The LES Smagorinsky turbulence model uses the Van Driest damping function close to the wall:

$$f_v = 1 - e^{-\frac{y^+}{A^+}}$$

7. **LES Dynamic Model.** The LES dynamic model is based on the model originally proposed by Germano et. al. (1991) and improved by Lilly (1992), with localization on Cs by Piomelli and Liu (1995).

8. **Thermal Law of the Wall.** When the thermal law of the wall is activated, the turbulent heat flux will be calculated as an additional output variable:

$$Q_t = \rho C_p \frac{U_\tau}{T_+}\,(T_s - T_c)$$

$$T_+ = \begin{cases} \mathrm{Pr}_t Y_+ & \text{if } Y_{+t} \le Y_{+tc} \\ \frac{\mathrm{Pr}_t}{\vartheta}\log(Y_+) + \left(3.85\mathrm{Pr}_t^{1.\overline{3}} - 1.3\right)^2 + 2.12\log(\mathrm{Pr}_t) & \text{otherwise} \end{cases}$$

*ICFD_CONTROL_TURB_SYNTHESIS

Purpose: This keyword enables the user impose a divergence-free turbulent field on inlets.

Card must be used jointly with VAD = 4 of keyword *ICFD_BOUNDARY_PRESCRIBED_VEL.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	IU	IV	IW	LS			
Type	I	F	F	F	F			
Default	0	10^{-3}	10^{-3}	10^{-3}	h_{min}			

VARIABLE	DESCRIPTION
PID	Part ID of the surface with the turbulent velocity inlet condition.
IU, IV, IW	Intensity of field fluctuations over x, y, and z directions, $$\text{IU} = \frac{u'}{u_{\text{avg}}}.$$
LS	Integral length scale of turbulence

Remarks:

1. If this card is not defined but a turbulent field inlet has been activated. See VAD = 4 of *ICFD_BOUNDARY_PRESCRIBED_VEL, the default parameters will be used.

*ICFD_DATABASE_AVERAGE

Purpose: This keyword enables the computation of time average variables at given time intervals.

Card 1	1	2	3	4	5	6	7	8
Variable	DT							
Type	F							
Default	none							

VARIABLE	DESCRIPTION
DT	Over each DT time interval, an average of the different fluid variables will be calculated and then reset when moving to the next DT interval.

Remarks:

1. The file name for this database is icfdavg.*.dat with the different averaged variable values copied in a ASCII format.

***ICFD_DATABASE_DRAG**_*{OPTION}*

Available options include

VOL

Purpose: This keyword enables the computation of drag forces over given surface parts of the model. If multiple keywords are given, the forces over the PID surfaces are given in separate files and are also added and output in a separate file.

For the VOL option, drag calculation can also be applied on a volume defined by ICFD_-PART_VOL. This is mostly useful in porous media applications to output the pressure drag of the porous media domain.

Surface Drag Cards. Include one card for each surface on which drag is applied. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	CPID	DTOUT	PEROUT	DIVI	ELOUT	SSOUT	
Type	I	I	F	I	I	I	I	
Default	none	none	0.	0	10	0	0	

VARIABLE	DESCRIPTION
PID	Part ID of the surface where the drag force will be computed.
CPID	Center point ID used for the calculation of the force's moment. By default the reference frame center is used is $\mathbf{0} = (0, 0, 0)$.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the ICFD timestep will be used.
PEROUT	Outputs the contribution of the different elements on the total drag in fractions of the total drag in the d3plots.
DIVI	Number of drag divisions for PEROUT. Default is 10 which means the contributions will be grouped in 10 deciles.
ELOUT	Outputs the drag value of each element in the d3plots.
SSOUT	Outputs the pressure loads caused by the fluid on each solid segment set in keyword format. FSI needs to be activated.

Remarks:

1. The file name for this database is icfdragi for instantaneous drag and icfdraga for the drag computed using average values of pressure and velocities.

2. The output contains:

   a) "Fpx" , "Fpy", and "Fpz" refer to the three components of the pressure drag force

   $$\mathbf{F}_p = \int P d\boldsymbol{A},$$

   where P is the pressure and A the surface area.

   b) "Fvx", "Fxy", and "Fvz" refer to the three components of the viscous drag force

   $$\mathbf{F}_v = \int \mu \frac{\partial \mathrm{u}}{\partial \mathrm{y}} \mathrm{d}\boldsymbol{A}.$$

   where $\frac{\partial \mathrm{u}}{\partial \mathrm{y}}$ is the shear velocity at the wall, μ is the viscosity and A is the surface area.

   c) "Mpx", "Mpy", "Mpz", "Mvx", "Mvy", and "Mvz" refer to the three components of the pressure and viscous force moments respectively.

*ICFD_DATABASE_FLUX

Purpose: This keyword enables the computation of the flow rate and average pressure over given parts of the model. If multiple keywords are given, separate files are output.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	DTOUT						
Type	I	F						
Default	none	↓						

VARIABLE	DESCRIPTION
PID	Part ID of the surface where the flow rates will be computed
DTOUT	Output frequency. Default is at every fluid timestep.

Remarks:

1. **Database Name.** The file name for this database is icfd_flux.dat.

2. **Database Components.** The flux database contains the flow rate through a section, called "output flux,"

$$\Phi = \sum_i (\mathbf{V}_i \cdot \mathbf{n}_i) A_i \ ,$$

the average pressure, called "Pre-avg,"

$$P_{\text{avg}} = \frac{\sum_i P_i A_i}{\sum_i A_i} \ ,$$

and the total area, called "Areatot."

*ICFD_DATABASE_FLUX_SURF

Purpose: Enable the computation of the flow rate and average pressure over a given surface (icfd part) of the model which is not necessary part of the analysis but lies in the fluid volume by mapping and interpolating volume results on the given surface.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	DTOUT						
Type	I	F						
Default	none	↓						

VARIABLE	DESCRIPTION
PID	Part ID of the surface where the flow rates will be computed
DTOUT	Output frequency. Default is at every fluid timestep.

Remarks:

1. **Database Name.** The file name for this database is icfd_fluxsurf.dat.

*ICFD_DATABASE_FORCE_DEM

Purpose: Enable the computation of the total fluid force that is transferred to the DEM particles over time. The output is in icfd_force_dem.dat.

Card 1	1	2	3	4	5	6	7	8
Variable	OUT							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
OUT	Flag to enable computing the fluid forces and generating output: EQ.0: No output is generated. EQ.1: Output is generated.

*ICFD_DATABASE_HTC

Purpose: This keyword allows the user to trigger the calculation of the Heat transfer coefficient using different methods and to control the output options.

Card 1	1	2	3	4	5	6	7	8
Variable	OUT	HTC	TB					OUTDT
Type	I	I	F					F
Default	0	0.	0.					0.

VARIABLE	DESCRIPTION
OUT	Determines if the solver should calculate the heat transfer coefficient and how to output it : EQ.0: No HTC calculation EQ.1: HTC calculated and output in LSPP as a surface variable. EQ.2: The solver will also look for FSI boundaries and output the HTC value at the solid nodes in an ASCII file called icfdhtci.dat. EQ.3: The solver will also look for FSI boundaries that are part of SEGMENT_SETS and output the HTC for those segments in an ASCII file called icfd_convseg.****.key in a format that can directly read by LS-DYNA for a subsequent pure structural thermal analysis.
HTC	Determines how the HTC is calculated. EQ.0: Automatically calculated by the solver based on the average temperature flowing through the pipe section (See Remark 1). EQ.1: User imposed value (See Remark 2).
TB	Value of the bulk temperature if HTC = 1.
OUTDT	Output frequency of the HTC in the various ASCII files. If left to 0., the solver will output the HTC at every timestep.

Remarks:

1. The heat transfer coefficient is frequently used in thermal applications to estimate the effect of the fluid cooling and it derived from a CFD calculation.

2. The heat transfer coefficient is defined as follows:

$$h = \frac{q}{T_s - T_b}$$

with q the heat flux, T_s the surface temperature and T_b the so called "bulk" temperature. For external aerodynamic applications, this bulk temperature is often defined as a constant (ambient or far field conditions, HTC = 1). However, for internal aerodynamic application, this temperature is often defined as an average temperature flowing through the pipe section with the flow velocity being used as a weighting factor (HTC = 0).

*ICFD_DATABASE_NODEAVG

Purpose: This keyword enables the computation of the average quantities on surface nodes defined in *ICFD_DATABASE_NODOUT.

Card 1	1	2	3	4	5	6	7	8
Variable	ON							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
ON	If equal to 1, the average quantities will be computed.

Remarks:

1. The file name for this database is icfd_nodeavg.dat.

*ICFD_DATABASE_NODOUT

Purpose: This keyword enables the output of ICFD data on surface nodes. For data in the fluid volume, it is advised to use points or tracers (See *ICFD_DATABASE_-POINTOUT).

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV	DTOUT						
Type	I	F						
Default	0	0.						

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	NID1	NID2	NID3	NID4	NID5	NID6	NID7	NID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the ICFD timestep will be used.
NID..	Node IDs.

Remarks:

1. The file name for this database is icfd_nodout.dat.

*ICFD_DATABASE_NTEMPOUT

Purpose: Output the temperature at individual nodes in a format consistent with *ICFD_-INITIAL_TEMPNODE to initialize a subsequent ICFD problem.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	NID	DTOUT						
Type	I	F						
Default	none	0.0						

VARIABLE	DESCRIPTION
NID	Internal ICFD node ID
DTOUT	Output frequency EQ.0.0: The ICFD timestep will be used.

*ICFD_DATABASE_POINTAVG

Purpose: This keyword enables the computation of the average quantities on point sets using the parameters defined in *ICFD_DATABASE_POINTOUT.

Card 1	1	2	3	4	5	6	7	8
Variable	ON							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
ON	If equal to 1, the average quantities will be computed.

Remarks:

1. The file name for this database is icfd_psavg.dat.

*ICFD_DATABASE_POINTOUT

Purpose: This keyword enables the output of ICFD data on points.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	PSID	DTOUT	PSTYPE	VX	VY	VZ		
Type	I	F	I	F	F	F		
Default	0	0.	0	0.	0.	0.		

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	PID	X	Y	Z				
Type	I	F	F	F				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
PSID	Point Set ID.
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the ICFD timestep will be used.
PSTYPE	Point Set type : EQ.0: Fixed points. EQ.1: Tracer points using prescribed velocity. EQ.2: Tracer points using fluid velocity. EQ.3: Tracer points using mesh velocity..
VX, VY, VZ	Constant velocities to be used when PSTYPE = 1
PID	Point ID

VARIABLE	DESCRIPTION
X, Y, Z	Point initial coordinates

Remarks:

1. The file name for this database is icfd_pointout.dat.

*ICFD_DATABASE_RESIDUALS

Purpose: This keyword allows the user to output the residuals of the various systems.

Card 1	1	2	3	4	5	6	7	8
Variable	RLVL							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
RLVL	Residual output level : EQ.0: No output. EQ.1: Only outputs the number of iterations needed for solving the pressure Poisson equation. EQ.2: Outputs the number of iterations for the momentum, pressure, mesh movement and temperature equations. EQ.3: Also gives the residual for each iteration during the solve of the momentum, pressure, mesh movement and temperature equations.

Remarks:

1. The file names for the momentum, pressure, mesh movement and temperature equations are called icfd_residuals.moms.dat, icfd_residuals.pres.dat, icfd_residuals.mmov.dat, and icfd_residuals.temp.dat respectively.

*ICFD_DATABASE_SSOUT

Purpose: Output the pressure load on a structure from the fluid. It can be useful for linear FSI applications, where the structure is made static, and the loads applied by the fluid are retrieved.

Card 1	1	2	3	4	5	6	7	8
Variable	OUT	OUTDT	LCIDSF					POFF
Type	I	I	I					F
Default	0	0.	0					0.

VARIABLE	DESCRIPTION
OUT	Determines if the solver should retrieve the pressure loads and how to output it: EQ.0: Inactive EQ.1: The fluid solver will collect the segment sets (see *SET_SEGMENT) that are part of a FSI boundary and retrieve the pressure for subsequent print out in an icfd_presseg and icfd_lcsegid pair of files. EQ.2: Same as 1 except the results are collected in a single icfd_presseg / icfd_plcsegid pair of files and the load curves associated with each segment are made functions of time, thus taking the transient nature of the CFD analysis into account. EQ.3: Same as 2 except uses a more memory efficient way of handling and outputting the data.
OUTDT	Frequency of the pressure extraction. If left as 0., the solver will extract the pressure of the fluid on the FSI boundary at every time step. This can lead to additional memory and calculation cost.
LCIDSF	Optional load curve ID to apply a scale factor on the fluid pressure output
POFF	Optional pressure offset on the fluid pressure output

*ICFD_DATABASE_SSOUT_EXCLUDE

Purpose: This keyword defines which segment set IDs are excluded from the SSOUT search. No forces coming from the fluid will be transmitted on those segment sets for output (see ICFD_DATABASE_SSOUT).

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	SSOUTID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
SSOUTID	Segment Set ID of the solid mechanics problem which is to be excluded from the output of the fluid forces on the solid boundaries.

*ICFD_DATABASE_TEMP

Purpose: This keyword enables the computation of the average temperature and the heat flux over given parts of the model. If multiple keywords are given, separate files are output.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	DTOUT						
Type	I	F						
Default	none	none						

VARIABLE	DESCRIPTION
PID	Part ID of the surface where the average temperature and heat flux will be computed.
DTOUT	Output frequency. Default is at every fluid timestep.

Remarks:

1. The file name for this database is icfd_thermal.dat.

2. Two average temperature are given in the icfd_thermal.dat file: "Temp-avg" and "Temp-sum". The average temperature is calculated using the local node area as weighting factor,

$$T_{\text{avg}} = \frac{\sum_i^N T_i A_i}{\sum_i^N A_i},$$

whereas, the sum is not weighted by area

$$T_{\text{sum}} = \frac{\sum_i^N T_i}{N}$$

If the mesh is regular, the two values will be of similar value. The icfd_thermal.dat output file also includes the average heat flux, the total surface area, and the average heat transfer coefficients (See *ICFD_DATABASE_HTC).

*ICFD_DATABASE_TIMESTEP

Purpose: This keyword enables the output of ICFD data regarding the ICFD timestep.

Output Options Card.

Card 1	1	2	3	4	5	6	7	8
Variable	OUTLV							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
OUTLV	Determines if the output file should be dumped. EQ.0: No output file is generated. EQ.1: The output file is generated.

Remarks:

1. The file name for this database is icfd_tsout.dat.

2. Outputs the run's ICFD timestep versus the timestep calculated using the ICFD CFL condition as criteria (autotimestep). This can be useful in cases using a fixed timestep where big mesh deformations and/or big fluid velocity changes occur in order to track how that fixed timestep value compares to the reference auto-timestep.

*ICFD_DATABASE_UINDEX

Purpose: This keyword allows the user to have the solver calculate the uniformity index (See Remark 1).

Card 1	1	2	3	4	5	6	7	8
Variable	OUT							
Type	I							
Default	0							

VARIABLE	DESCRIPTION
OUT	Determines if the solver should calculate the uniformity index. EQ.0: Off. EQ.1: On.

Remarks:

1. **Uniformity Index.** The uniformity index is a post treatment quantity which measures how uniform the flow is through a given section. It is especially useful in internal aerodynamics cases. It is expressed as :

$$\gamma = 1 - \frac{1}{2nA}\sum_{i=1}^{n}\left[\frac{\sqrt{(u_i - \bar{u})^2}}{\bar{u}}A_i\right]$$

with A_i, the local cell area, A the total section area, u_i the local velocity, $\bar{u}$ the average velocity through the section, and n the number of cells.

Values close to 0 means that the flow is very unevenly distributed. This can be used to identify bends, corners or turbulent effects. Values close to 1 imply smooth or equally distributed flow through the surface.

*ICFD_DATABASE_WETNESS

Purpose: For free surface problems, output the wetted area as a function of time for a given surface part. It is output to an ASCII file beginning with icfd_wetness.

Card 1	1	2	3	4	5	6	7	8
Variable	SPID	DTOUT						
Type	I	F						
Default	none	0.						

VARIABLE	DESCRIPTION
SPID	Part ID of the surface where the wetted area will be computed
DTOUT	Time interval to print the output. If DTOUT is equal to 0.0, then the ICFD time step will be used.

Remarks:

1. **Wetted Area.** The ICFD solver identifies a surface as wet if its level set value is positive.

*ICFD_DEFINE_HEATSOURCE

Purpose: This keyword defines a volumetric heat source for the heat equation solve.

Card 1	1	2	3	4	5	6	7	8
Variable	HSID	LCID	SHAPE	R	PTID1	PTID2		
Type	I	I	I	F	I	I		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
HSID	Heat source ID.
LCID	Load curve ID specifying the evolution of the heat source term function of time for the X, Y and Z dofs, see *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
SHAPE	Shape of the volumetric heat source: EQ.1 : Box shape EQ.2 : Cylinder shape EQ.3 : Sphere shape
R	Radius of the sphere is SHAPE = 3
PTID1	ID of point (See ICFD_DEFINE_POINT) of minimum coordinates if SHAPE = 1, tail point if SHAPE = 2, origin if SHAPE = 3.
PTID2	ID of point of maximum coordinates if SHAPE = 2, head point if SHAPE = 2.

.

*ICFD_DEFINE_RESIDENCETIMESOURCE

Purpose: Specify a volumetric residence time (RT) source.

Card 1	1	2	3	4	5	6	7	8
Variable	RTSID	ISHAPE	R	PTID1	PTID2	MASSDIFF	DEATHT	IRT0PBC
Type	I	I	F	I	i	F	F	I
Default	none	none	none	none	none	10^{-6}	↓	0

VARIABLE	DESCRIPTION
RTSID	RT source ID
ISHAPE	Shape of the volumetric RT source: EQ.1: Box EQ.2: Cylinder EQ.3: Sphere
R	Radius of the cylinder if ISHAPE = 2 or radius of the sphere if ISHAPE = 3
PTID1	ID of a point (see *ICFD_DEFINE_POINT) giving the minimum coordinate of the box if ISHAPE = 1, the tail point for the cylinder if ISHAPE = 2, or the origin of the sphere if ISHAPE = 3
PTID2	ID of a point giving the maximum coordinate of the box if ISHAPE = 1 or the head point of the cylinder if ISHAPE = 2.
MASSDIFF	Mass diffusion for the transport equation
DEATHT	End time for the source. EQ.0.0: End time of the simulation
IRT0PBC	Flag for which prescribed boundaries RT = 0 is imposed: EQ.0: Imposed only on boundaries with prescribed velocity EQ.1: Imposed on boundaries with either prescribed velocity or prescribed pressure

*ICFD_DEFINE_SOURCE

Purpose: Define a volumetric external force for the momentum equation solve.

Card 1	1	2	3	4	5	6	7	8
Variable	SID	LCIDX	LCIDY	LCIDZ	SHAPE	R	PTID1	PTID2
Type	I	I	I	I	I	F	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
SID	Source ID
LCIDX/Y/Z	Load curve IDs specifying the evolution of the volumetric force as a function of time for the three global components
SHAPE	Shape to which the volumetric force is applied: EQ.1: Box EQ.2: Cylinder EQ.3: Sphere
R	Radius of the cylinder or sphere if SHAPE = 2 or 3
PTID1	Point ID (see *ICFD_DEFINE_POINT) whose meaning depends on SHAPE: SHAPE.EQ.1: Minimum coordinates of the box SHAPE.EQ.2: Tail point of the cylinder SHAPE.EQ.3: Origin of the sphere
PTID2	Point ID (see *ICFD_DEFINE_POINT) whose meaning depends on SHAPE: SHAPE.EQ.1: Maximum coordinates of the box SHAPE.EQ.2: Head point of the cylinder

*ICFD_DEFINE_SPTRANSPSOURCE

Purpose: Specify a volumetric species source for the species transport solver.

Card 1	1	2	3	4	5	6	7	8
Variable	SPTRSID	LCID	ISHAPE	R	PTID1	PTID2	MASSDIFF	DEATHT
Type	I	I	I	F	I	i	F	F
Default	none	none	none	none	none	none	10^{-6}	↓

VARIABLE	DESCRIPTION
SPTRSID	Species transport source ID
LCID	Load curve ID specifying the evolution of the species source term as a function of time for the X, Y, and Z degrees of freedom (see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, and *DEFINE_FUNCTION). If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y,z, vx, vy, vz, temp, pres, time).
ISHAPE	Shape of the volumetric species source: EQ.1: Box EQ.2: Cylinder EQ.3: Sphere
R	Radius of the cylinder if ISHAPE = 2 or radius of the sphere if ISHAPE = 3
PTID1	ID of a point (see *ICFD_DEFINE_POINT) giving the minimum coordinate of the box if ISHAPE = 1, the tail point for the cylinder if ISHAPE = 2, or the origin of the sphere if ISHAPE = 3
PTID2	ID of a point giving the maximum coordinate of the box if ISHAPE = 1 or the head point of the cylinder if ISHAPE = 2.
MASSDIFF	Mass diffusion for the transport equation
DEATHT	End time for the source. EQ.0.0: End time of the simulation

*ICFD_DEFINE_TURBSOURCE

Purpose: This keyword defines a external source for the RANS turbulent equations.

Card 1	1	2	3	4	5	6	7	8
Variable	TSID	LCIDK	LCIDEP	LCIDNU	SHAPE	R	PTID1	PTID2
Type	I	I	I	I	I	F	I	I
Default	none	none	none	none	none	none	none	None

VARIABLE	DESCRIPTION
TSID	Turbulent external source ID.
LCIDK	Load curve ID specifying the evolution of the external source term function of time for the turbulent kinetic energy k equation, see *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
LCIDEP	Load curve ID specifying the evolution of the external source term function of time for the turbulent diffusion ε or specific rate of dissipation w equation, see *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
LCIDNU	Load curve ID specifying the evolution of the external source term function of time for the kinematic eddy turbulent viscosity equation used in the Spalart-Allmaras model, see *DEFINE_CURVE,*DEFINE_CURVE_FUNCTION or *DEFINE_FUNCTION. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.

VARIABLE	DESCRIPTION
SHAPE	Shape of the external source: EQ.1 : Box shape EQ.2 : Cylinder shape EQ.3 : Sphere shape
R	Radius of the sphere is SHAPE = 3
PTID1	ID of point (See ICFD_DEFINE_POINT) of minimum coordinates if SHAPE = 1, tail point if SHAPE = 2, origin if SHAPE = 3.
PTID2	ID of point of maximum coordinates if SHAPE = 2, head point if SHAPE = 2.

.

*ICFD_DEFINE_POINT

Purpose: This keyword defines a point in space that could be used for multiple purposes.

Card 1	1	2	3	4	5	6	7	8
Variable	POID	X	Y	Z	CONSTPID			
Type	I	F	F	F	I			
Default	none	none	none	none	none			

Optional Card 2. Load curve IDS specifying velocity components of translating point

Card 2	1	2	3	4	5	6	7	8
Variable	LCIDX	LCIDY	LCIDZ					
Type	I	I	I					
Default	0	0	0					

Optional Card 3. Load curve IDS and rotation axis of rotating point

Card 2	1	2	3	4	5	6	7	8
Variable	LCIDW	XT	YT	ZT	XH	YH	ZH	
Type	I	F	F	F	F	F	F	
Default	0	none	none	none	none	none	none	

VARIABLE	DESCRIPTION
POID	Point ID.
X/Y/Z	x, y ,z coordinates for the point.

VARIABLE	DESCRIPTION
CONSTPID	Surface Part ID to which the point is constrained. This means that if the selected surface moves, then the localization of the point will update as well.
LCIDX/LCIDY/LCIDZ	The point can be made to translate. Those are the three load curve IDs for the three translation components.
LCIDW	The point can also be made to rotate. This load curve specifies the angular velocity.
XT/YT/ZT	Rotation axis tail point coordinates.
XH/YH/ZH	Rotation axis head point coordinates.

.

*ICFD_DEFINE_NONINERTIAL

Purpose: This keyword defines a non-inertial reference frame in order to avoid heavy mesh distortions and attendant remeshing associated with large-scale rotations. This is used to model, for example, spinning cylinders, wind turbines, and turbo machinery.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	W1	W2	W3	R	PTID	L	LCID	RELV
Type	F	F	F	F	I	F	I	I
Default	none	none	none	none	none	none	none	0

VARIABLE	DESCRIPTION
W1, W2, W3	Rotational Velocity along the X,Y,Z axes
R	Radius of the rotating reference frame. If a negative value if given, then the absolute value will refer to a *DEFINE_FUNCTION ID. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.
PTID	Starting point ID for the reference frame (See *ICFD_DEFINE_POINT)
L	Length of the rotating reference frame

LCID Load curve for scaling factor of w. If a negative value is entered, then the absolute value will refer to a *DEFINE_FUNCTION ID. If a DEFINE_FUNCTION is used, the following parameters are allowed: $f(x, y, z, vx, vy, vz, temp, pres, time)$.

Figure

VARIABLE	DESCRIPTION
RELV	Velocities computed and displayed: EQ.0: Relative velocity, only the non-rotating components of the velocity are used and displayed. EQ.1: Absolute velocity . All the components of the velocity are used. Useful in cases where several or at least one non-inertial reference frame is combined with an inertial "classical" reference frame.

*ICFD_DEFINE_WAVE_DAMPING

Purpose: This keyword defines a damping zone for free surface waves.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	NID	L	F1	F2	N	LCID	
Type	I	I	F	F	F	I	I	
Default	none	none		10	10	1	none	

VARIABLE	DESCRIPTION
PID	Point ID defining the start of the damping layer.
NID	Normal ID defined using ICFD_DEFINE_POINT and pointing to the outgoing direction of the damping layer.
L	Length of damping layer. If no is value specified, the damping layer will have a length corresponding to five element lengths.
F1/F2	Linear and quadratic damping factor terms.
N	Damping term factor.
LCID	Load curve ID acting as temporal scale factor on damping term.

Remarks:

1. The damping is achieved by adding a source term to the momentum equations :

$$s^d = w\ (f_1 + f_2|u|)\ u$$

with w the weight function :

$$w = \frac{e^{\gamma} - 1}{e - 1}$$

and γ the blending function which allows a smooth insertion of the source term in the damping layer :

$$\gamma = \left(\frac{x - x_{sd}}{x_{ed} - x_{sd}}\right)^n$$

x_{sd} and x_{ed} representing the start and end coordinates of the damping zone.

*ICFD_INITIAL

Purpose: Simple initialization of velocity and temperature within a volume.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	Vx	Vy	Vz	T	P		DFUNC
Type	I	F	F	F	F	F		I
Default	none	none	none	none	none	none		0

VARIABLE	DESCRIPTION
PID	Part ID for the volume elements or the surface elements where the values are initialized (see *ICFD_PART_VOL and *ICFD_PART). PID = 0 to assign the initial condition to all nodes at once.
Vx	x coordinate for the velocity.
Vy	y coordinate for the velocity.
Vz	z coordinate for the velocity.
T	Initial temperature.
P	Initial Pressure.
DFUNC	Option to define initial conditions using *DEFINE_FUNCTION EQ.0: Turned off. EQ.1: Turned on. All previous flags for initial velocity, pressure and temperature now refer to *DEFINE_FUNCTION IDs. The following parameters are allowed : $f(x,y,z)$, allowing to define initial profiles function of coordinates.

*ICFD_INITIAL_LEVELSET

Purpose: Define an initial level set surface instead of a multi-fluid domain (replaces the need for *MESH_INTERF).

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	STYPE	NX	NY	NZ	X	Y	Z	INVERT
Type	I	F	F	F	F	F	F	I
Default	none	none	none	none	none	none	none	0

VARIABLE	DESCRIPTION
STYPE	Initial surface type: EQ.0/1: Generated by a section plane. EQ.2: Generated by a box. See Remark 1. EQ.3: Generated by a sphere. EQ.4: Generated by a cylinder.
NX, NY, NZ	X, Y and Z components of the section plane normal if STYPE = 1. Minimum coordinates of the box, P_{min}, if STYPE = 2. NX is the sphere/cylinder radius if STYPE = 3 and STYPE = 4. NY is the cylinder length if STYPE = 4. NZ is the global axis if STYPE = 4 (NZ = 1, 2, or 3 means X-axis, Y-axis, or Z-axis, respectively).
X, Y, Z	X, Y and Z components of the section plane origin point if STYPE = 1 and 4. Maximum coordinates of the box, P_{max}, if STYPE = 2. Coordinates of the sphere origin point if STYPE = 3.
INVERT	Inversion of initial level set: EQ.0: No inversion. Positive level set values are assigned to nodes contained within the volume defined by STYPE. EQ.1: The sign of the initial level set values is reversed.

Remarks:

1. **Box Adjacent to Fluid Boundaries.** When STYPE = 2 is used and the box is adjacent to the fluid boundaries such as during a dam break simulation, the distance from any point in the fluid to the fluid boundary must remain smaller than the distance to the defined box. Therefore, the P_{min} coordinates need to be defined far outside the initial fluid domain.

2. **Multiple Keyword Definitions.** Multiple definitions of this keyword are possible, but the different shapes generated must not intersect for a correct initialization.

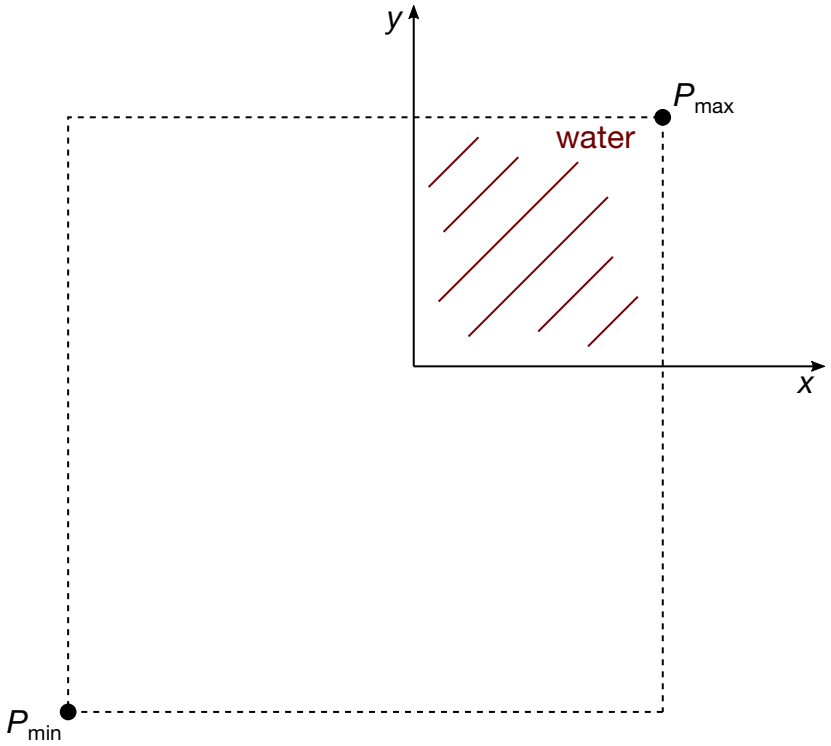


Figure 7-3. 2D Dam breaking example with initial levelset surface defined using STYPE = 2. P_{min} is defined sufficiently far away from the fluid surface boundaries.

*ICFD_INITIAL_SPTRANSP

Purpose: Initialize the concentration of the species being transported within a volume.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	CONC	IDFUNC					
Type	I	F	I					
Default	0	none	0					

VARIABLE	DESCRIPTION
PID	Part ID for the volume elements or surface elements where the values are initialized. EQ.0: Assign the initial condition to all nodes at once.
CONC	Initial concentration. CONC is *DEFINE_FUNCTION ID if IDFUNC = 1.
IDFUNC	Flag to define initial conditions using a *DEFINE_FUNCTION: EQ.0: Turned off. EQ.1: Turned on. CONC is a *DEFINE_FUNCTION ID. The following parameters are allowed: f(x, y, z). This allows for defining initial profiles as a function of coordinates.

*ICFD_INITIAL_TEMPNODE

Purpose: Allow the solver to initialize the temperature at individual nodes.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	NID	TEMP						
Type	I	F						
Default	none	none						

VARIABLE	DESCRIPTION
NID	Internal ICFD node ID
TEMP	Initial temperature value

*ICFD_INITIAL_TURBULENCE

Purpose: Modify the default initial values of the turbulence quantities for a RANS turbulence model.

Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	I	R	K	EW			
Type	I	F	F	F	F			
Default	none	none	none	optional	optional			

VARIABLE	DESCRIPTION
PID	Part ID for the volume elements or the surface elements where the values are initialized (see *ICFD_PART_VOL and *ICFD_PART). Set PID to 0 to assign the initial condition to all nodes at once.
I	Initial turbulence intensity, I
R	Initial turbulence viscosity to laminar viscosity ratio ($r = \mu_{\text{turb}}/\mu$).
K	Initial kinetic energy. When defined, it replaces the choice of I. LT.0.0: |K| refers to a *DEFINE_FUNCTION ID. The following parameters are allowed: $f(x,y,z)$, allowing you to define initial profiles as a function of coordinates.
EW	Initial turbulence specific dissipation rate or dissipation rate depending on the choice of turbulence model. When defined, it replaces the choice of R. LT.0.0: |EW| refers to a *DEFINE_FUNCTION ID. The following parameters are allowed: $f(x,y,z)$, allowing you to define initial profiles as a function of coordinates.

Remarks:

1. **Default Initial Conditions.** If no initial conditions have been assigned to a specific PID, the solver will automatically pick I = 0.05 (5%) and R = 10000.

***ICFD_MAT**_*{OPTION}*

Available options include

TITLE

Purpose: Specify physical properties for the fluid material.

Fluid Material Card Sets:

The Material Fluid Parameters Card is required. If a second card is given, it must be a Thermal Fluid Parameters Card. If the fluid thermal properties are not needed, the second card can be a blank card. With the third card, you can associate the fluid material to a non-Newtonian model, to a porous media model, and/or to a species transport model (see *ICFD_MODEL_NONNEWT, *ICFD_MODEL_POROUS, and *ICFD_MODEL_SPECIES_TRANSPORT).

Material Fluid Parameters Card.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	FLG	RO	VIS	ST	STSFLCID	CA	
Type	I	I	F	F	F	I	F	
Default	none	1	0.	0.	0.	none	0.	

Thermal Fluid Parameters Card. Only to be defined if the thermal problem is solved. Otherwise include a blank card.

Card 2	1	2	3	4	5	6	7	8
Variable	HC	TC	BETA	PRT	HCSFLCID	TCSFLCID		
Type	F	F	F	F	I	I		
Default	0.	0.	0.	0.85	none	none		

Additional fluid models. Only to be defined if the fluid is non-Newtonian, there is a porous media, and/or a species is being transported.

Card 3	1	2	3	4	5	6	7	8
Variable	NNMOID	PMMOID		SPTRID				
Type	I	I		I				
Default	optional	optional		optional				

VARIABLE	DESCRIPTION
MID	Material ID
FLG	Flag to choose between fully incompressible, slightly compressible, or barotropic flows: EQ.0: Vacuum (free surface problems only) EQ.1: Fully incompressible fluid.
RO	Flow density
VIS	Dynamic viscosity
ST	Surface tension coefficient
STSFLCID	Load curve ID for scale factor applied on ST as a function of time. See *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_FUNCTION. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time).
CA	Contact angle
HC	Heat capacity
TC	Thermal conductivity
BETA	Thermal expansion coefficient used in the Boussinesq approximation for buoyancy
PRT	Turbulent Prandlt number. Only used if K-Epsilon turbulence model selected.

VARIABLE	DESCRIPTION
HCSFLCID	Load curve ID for scale factor applied on HC function of time. See *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_-FUNCTION. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time).
TCSFLCID	Load curve ID for scale factor applied on TC function of time. See *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_-FUNCTION. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time).
NNMOID	Non-Newtonian model ID. This refers to a Non-Newtonian fluid model defined using *ICFD_MODEL_NONNEWT.
PMMOID	Porous media model ID. This refers to a porous media model defined using *ICFD_MODEL_POROUS.
SPTRID	Species transport model ID. This refers to a species transport model defined using *ICFD_MODEL_SPECIES_TRANSPORT.

*ICFD_MODEL_NONNEWT

Purpose: Specify a non-newtonian model or a viscosity law that can associated to a fluid material.

Non-Newtonian Model ID and type.

Card 1	1	2	3	4	5	6	7	8
Variable	NNMOID	NNID						
Type	I	I						
Default	none	none						

Non-Newtonian Fluid Parameters Card.

Card 2	1	2	3	4	5	6	7	8
Variable	K	N	MUMIN	LAMBDA	ALPHA	TALPHA		
Type	F	F	F	F	F	F		
Default	0.0	0.0	0.0	1.e30	0.0	0.0		

VARIABLE	DESCRIPTION
NNMOID	Non-Newtonian Model ID.
NNID	Non-Newtonian fluid model type : EQ.1 : Power-Law model EQ.2 : Carreau model EQ.3 : Cross model EQ.4 : Herschel-Bulkley model EQ.5 : Cross II model EQ.6 : Sutherland formula for temperature dependent viscosity EQ.7 : Power-Law for temperature dependent viscosity EQ.8 : Viscosity defined by Load Curve ID or Function ID

VARIABLE	DESCRIPTION
K	Consistency index if NNID = 1 and 4. Zero shear Viscosity if NNID = 2,3 and 5.Reference viscosity if NNID = 6 and NNID = 7. Load curve ID or function ID if NNID = 8.
N	Measure of the deviation of the fluid from Newtonian (Power Law index) for NNID = 1,2,3,4,5,7. Not used for NNID = 6 and 8.
MUMIN	Minimum acceptable viscosity value if NNID = 1. Infinite Shear Viscosity if NNID = 2,5.Yielding viscosity if NNID = 4.Not used if NNID = 3,6,7,8.
LAMBDA	Maximum acceptable viscosity value if NNID = 1. Time constant if NNID = 2, 3, 5. Yield Stress Threshold if NNID = 4.Sutherland constant if NNID = 6. Not used if NNID = 7,8.
ALPHA	Activation energy if NNID = 1, 2. Not used if NNID = 3,4,5,6,7,8.
TALPHA	Reference temperature if NNID = 2. Not used if NNID = 1,3,4,5,6,7,8

Remarks:

1. For the Non-Newtonian models, the viscosity is expressed as :

   a) POWER-LAW :

$$\mu = k\dot{\gamma}^{n-1}e^{\alpha T_0/T}$$

$$\mu_{min} < \mu < \mu_{max}$$

   With k the consistency index, n the power law index, α the activation energy, T_0 the initial temperature, T the temperature at any given time t, μ_{min} the minimum acceptable viscosity and μ_{max} the maximum acceptable viscosity.

   b) CARREAU :

$$\mu = \mu_\infty + (\mu_0 - \mu_\infty)\left[1 + (H(T)\dot{\gamma}\lambda)^2\right]^{(n-1)/2}$$

$$H(T) = exp\left[\alpha\left(\frac{1}{T - T_0} - \frac{1}{T_\alpha - T_0}\right)\right]$$

   With μ_∞ the infinite shear viscosity, μ_0 the zero shear viscosity, n the power law index, λ a time constant, α the activation energy, T_0 the initial temperature, T

the temperature at any given time t and T_α the reference temperature at which $H(T) = 1$.

c) CROSS :

$$\mu = \frac{\mu_0}{1 + (\lambda\dot{\gamma})^{1-n}}$$

With μ_0 the zero shear viscosity, n the power law index and λ a time constant.

d) HERSCHEL-BULKLEY :

$$\mu = \mu_0 \; if \; (\dot{\gamma} < \tau_0/\mu_0)$$

$$\mu = \frac{\tau_0 + k[\dot{\gamma}^n - (\tau_0/\mu_0)^n]}{\dot{\gamma}}$$

With k the consistency index, τ_0 the Yield stress threshold, μ_0 the yielding viscosity and n the power law index.

e) CROSS II :

$$\mu = \mu_\infty + \frac{\mu_0 - \mu_\infty}{1 + (\lambda\dot{\gamma})^n}$$

With μ_0 the zero shear viscosity, μ_∞ the infinite shear viscosity, n the power law index and λ a time constant.

2. For the temperature dependent viscosity models, the viscosity is expressed as :

a) SUTHERLAND's LAW :

$$\mu = \mu_0 (\frac{T}{T_0})^{3/2} \frac{T_0 + S}{T + S}$$

With μ_0 a reference viscosity, T_0 the initial temperature (which therefore must not be 0.), T the temperature at any given time t and S Sutherland's constant.

b) POWER LAW :

$$\mu = \mu_0 (\frac{T}{T_0})^n$$

With μ_0 a reference viscosity, T_0 the initial temperature (which therefore must not be 0.), T the temperature at any given time t and n the power law index.

3. For NNID = 8, a load curve function of time, a curve function or a function can be used. If it references a DEFINE_FUNCTION, the following arguments are allowed $f(x, y, z, vx, vy, vz, temp, pres, shear, time)$.

*ICFD_MODEL_POROUS

Purpose: Specify a porous media model.

Card Summary:

Card 1. This card is required.

PMMOID	PMMTYPE	FORM	RHOCP	KAPPA			

Card 2a. This card is included if PMMTYPE = 1, 2, or 8.

POR	PER	FF		PSFLCID			

Card 2b. This card is included if PMMTYPE = 3 or 10.

POR	TH		FABTH	PVLCID			

Card 2c. This card is included if PMMTYPE = 4, 6, or 7.

POR							

Card 2d. This card is included if PMMTYPE = 5.

POR	THX	THY	THZ	PVLCIDX	PVLCIDY	PVLCIDZ	

Card 2e. This card is included if PMMTYPE = 11.

POR	ALPHA	BETA					

Card 3. This card is included if PMMTYPE = 4, 5, 6, or 7

KXP	KYP	KZP					

Card 4a. This card is included if PMMTYPE = 4 or 6.

PROJXPX	PROJXPY	PROJXPZ	PROJYPX	PROJYPY	PROJYPZ		

Card 4b. This card is included if PMMTYPE = 5 or 7.

PID1REF	PID2REF						

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	PMMOID	PMMTYPE	FORM	RHOCP	KAPPA			
Type	I	I	I	F	F			
Default	none	none	0	0.0	0.0			

VARIABLE	DESCRIPTION
PMMOID	Porous media model ID
PMMTYPE	Porous media model type: EQ.1: Isotropic porous media - Ergun correlation EQ.2: Isotropic porous media - Darcy-Forchheimer model EQ.3: Isotropic porous media - permeability defined through pressure-velocity data EQ.4: Anisotropic porous media. Fixed local reference frame (see Figure 7-4). EQ.5: Anisotropic porous media model - moving local reference frame and permeability vector in local reference frame (x', y', z') defined by three pressure-velocity curves. EQ.6: Anisotropic porous media model - moving local reference frame and permeability vector constant. EQ.7: Anisotropic porous media model - moving local reference frame and permeability vector constant. This model differs from PMMTYPE = 6 in the way the local reference frame is moved. EQ.8: Main parachute model to be used jointly with *MESH_EMBEDSHELL for the parachute surface. Similar to PMMTYPE = 2. EQ.10: Parachute model to be used jointly with *MESH_EMBEDSHELL where the fabric permeability and Forchheimer factor are computed from the pressure-velocity curves of experimental data given by a *LOAD_CURVE. Similar to PMMTYPE = 3.

VARIABLE	DESCRIPTION
	EQ.11: Parachute model similar to PMMTYPE = 8, but pressure gradient is directly defined by coefficients α and β as: $\frac{\Delta P(u_x)}{\Delta x} = \alpha u_x + \beta {u_x}^2$.
FORM	Porous media formulation: EQ.0: Classical (default) EQ.2: Interstitial velocity
RHOCP	Density of the structure multiplied by the specific heat of the structure
KAPPA	Thermal conductivity of the structure

Porous Media Parameters Card (PMMTYPE = 1, 2, and 8). This card is included PMMTYPE = 1, 2, or 8.

Card 2a	1	2	3	4	5	6	7	8
Variable	POR	PER	FF		PSFLCID			
Type	F	F	F		I			
Default	0.	0.	0.		optional			

VARIABLE	DESCRIPTION
POR	Porosity, ε
PER	Permeability, κ
FF	Forchheimer factor to be defined if PMMTYPE = 2 or 8.
PSFLCID	Optional permeability scale factor load curve ID, *DEFINE_CURVE_FUNCTION ID or *DEFINE_FUNCTION ID. If a *DEFINE_FUNCTION is used, the following parameters are allowed: f(x, y, z, vx, vy, vz, temp, pres, time).

Porous Media Parameters Card (PMMTYPE = 3 and 10). This card is included if PMMTYPE = 3 or 10.

Card 2b	1	2	3	4	5	6	7	8
Variable	POR	TH		FABTH	PVLCID			
Type	F	F		F	I			
Default	0.	0.		0.	none			

VARIABLE	DESCRIPTION
POR	Porosity, ε
TH	Probe thickness if PMMTYPE = 3
FABTH	Fabric thickness if PMMTYPE = 10
PVLCID	Pressure as a function of velocity load curve ID

Porous Media Parameters Card (PMMTYPE = 4, 6, and 7). This card is included if PMMTYPE = 4, 6, or 7.

Card 2c	1	2	3	4	5	6	7	8
Variable	POR							
Type	F							
Default	0.							

VARIABLE	DESCRIPTION
POR	Porosity, ε

Porous Media Parameters Card (PMMTYPE = 5). This card is included if PMMTYPE = 5.

Card 2d	1	2	3	4	5	6	7	8
Variable	POR	THX	THY	THZ	PVLCIDX	PVLCIDY	PVLCIDZ	
Type	F	F	F	F	I	I	I	
Default	0.	0.	0.	0.	none	none	none	

VARIABLE	DESCRIPTION
POR	Porosity, ε
THX	Probe thickness, Δx
THY	Probe thickness, Δy
THZ	Probe thickness, Δz
PVLCIDX	Load curve ID for pressure as a function of velocity in the global X-direction
PVLCIDY	Load curve ID for pressure as a function of velocity in the global Y-direction
PVLCIDZ	Load curve ID for pressure as a function of velocity in the global Z-direction

Porous Media Parameters Card (PMMTYPE = 11). This card is included if PMMTYPE = 11.

Card 2e	1	2	3	4	5	6	7	8
Variable	POR	ALPHA	BETA					
Type	F	F	F					
Default	0.	0.	0.					

VARIABLE	DESCRIPTION
POR	Porosity, ε
ALPHA	Coefficient, α
BETA	Coefficient, β

Permeability Vector Card in local reference frame. Only to be defined if the porous media is anisotropic (PMMTYPE = 4, 5, 6, 7).

Card 3	1	2	3	4	5	6	7	8
Variable	KXP	KYP	KZP					
Type	F	F	F					
Default	0.	0.	0.					

VARIABLE	DESCRIPTION
KXP, KYP, KZP	Permeability vector in local reference frame (x', y', z'). Those values become scale factors if PMMTYPE = 5.

Projection of Local Vectors in Global Reference Frame. This card is defined if PMMTYPE = 4 or 6.

Card 4a	1	2	3	4	5	6	7	8
Variable	PROJXPX	PROJXPY	PROJXPZ	PROJYPX	PROJYPY	PROJYPZ		
Type	F/I	F/I	F/I	F/I	F/I	F/I		
Default	0./0	0./0	0./0	0./0	0./0	0./0		

VARIABLE	DESCRIPTION
PROJXPX, PROJXPY, PROJXPZ	Projection of the local permeability vector, **x**′, in the global reference frame, (x, y, z). If PMMTYPE = 6, PROJXPX, PROJXPY, and PRPJXPZ become load curve IDs, so the coordinates of the local **x**′ vector can change in time.

VARIABLE	DESCRIPTION
PROJYPX, PROJYPY, PROJYPZ	Projection of the local permeability vector, **y**′, in the global reference frame, (x, y, z). If PMMTYPE = 6, PROJYPX, PROJYPY, and PRPJYPZ become load curve IDs, so the coordinates of the local **y**′ vector can change in time.

Local Reference Frame Vectors. This card is defined if PMMTYPE = 5 or 7.

Card 4b	1	2	3	4	5	6	7	8
Variable	PID1REF	PID2REF						
Type	I	I						
Default	0	0.						

VARIABLE	DESCRIPTION
PID1REF, PID2REF	Two local reference frame vectors are defined by the coordinates of the two-point IDs defined by PID1REF and PID2REF. (See *ICFD_DEFINE_POINT). Since those points can be made to move, it is therefore possible to define a moving reference frame for the anisotropic porous media domain.

Remarks:

1. **Generalized Flow Equations in a Porous Media.** Let ε be the porosity and κ be the permeability of the porous media. Then,

$$\varepsilon = \frac{\text{void volume}}{\text{total volume}} \ .$$

u_i, the volume averaged velocity field, can then be defined in terms of the fluid velocity field, u_{if}, as:

$$u_i = \varepsilon u_{if} \ .$$

The generalized flow equations of momentum and mass conservation can be expressed as:

$$\frac{\partial u_i}{\partial x_i} = 0$$

$$\frac{\rho}{\varepsilon}\left[\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\frac{u_i u_j}{\varepsilon}\right)\right] = -\frac{1}{\varepsilon}\frac{\partial (P\varepsilon)}{\partial x_i} + \frac{\mu}{\varepsilon}\left(\frac{\partial^2 u_i}{\partial x_j \partial x_j}\right) + \rho g_i - D_i$$

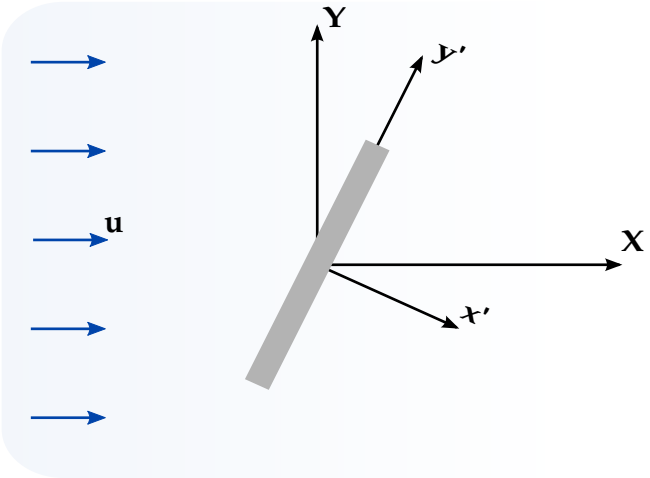


Figure 7-4. Anisotropic porous media vectors definition (PMMTYPE = 4, 5, 6, and 7). The vectors **X** and **Y** are the global axes; $\mathbf{x}'$ and $\mathbf{y}'$ define the system for the primed coordinate(x', y', z').

where D_i are the forces exerted on the fluid by the porous matrix (see Remarks 2 and 3).

2. **Porous Forces for Isotropic Models.** For the isotropic model, the porous forces are a function of the matrix porosity and its permeability. For the isotropic case, three models are available:

   a) *Model 1 (Ergun correlation).*

$$D_i = \frac{\mu u_i}{\kappa} + \frac{1.75\rho|U|}{\sqrt{150}\sqrt{\kappa}\varepsilon^{3/2}} u_i$$

   b) *Model 2 (Darcy-Forcheimer).*

$$D_i = \frac{\mu u_i}{\kappa} + \frac{F\varepsilon\rho|U|}{\sqrt{\kappa}} u_i$$

   c) *Model 3.* Using the $\Delta P - V$ experimental data. In this case, it is assumed that the pressure-velocity curve was obtained by applying a pressure difference or pressure drop on both ends of a porous slab of thickness Δx with porous properties κ and ε. It then becomes possible for the solver to fit that experimental curve with a quadratic polynomial of the form $\Delta P(u_x) = \alpha {u_x}^2 + \beta u_x$. Once α and β are known, it is possible to estimate D_i.

3. **Anisotropic Darcy-Forcheimer Term.** The anisotropic (see Figure 7-4) version of the Darcy-Forcheimer term can be written as:

$$D_i = \mu B_{ij} \mu_j + F\varepsilon|U|C_{ij}u_j$$
$$B_{ij} = \left(K_{ij}\right)^{-1}$$
$$C_{ij} = \left(K_{ij}\right)^{-1/2}$$

Here K_{ij} is the anisotropic permeability tensor.

*ICFD_MODEL_SPECIES_TRANSPORT

Purpose: Specify a species transport model that can be associated with a fluid material.

Species transport model ID and type.

Card 1	1	2	3	4	5	6	7	8
Variable	SPTRID	SPTRTYPE						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
SPTRID	Species transport model ID
SPTRTYPE	Species transport model type: EQ.1: Passive species transport

***ICFD_PART**_*{OPTION}*

Available options include

TITLE

Purpose: Define parts for this incompressible flow solver.

The TITLE option allows the user to define an additional optional line with a HEADING in order to associate a name to the part.

Card 1	1	2	3	4	5	6	7	8
Variable	HEADING							
Type	A							
Default	none							

Part Material Card. Include as many cards as needed. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	PID	SECID	MID					
Type	I	I	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
PID	Part identifier for fluid surfaces.
SECID	Section identifier defined with the *ICFD_SECTION card.
MID	Material identifier defined with the *ICFD_MAT card.

*ICFD_PART_VOL_{OPTION}

Available options include

TITLE

Purpose: This keyword assigns material properties to the nodes enclosed by surface ICFD parts.

The TITLE option allows the user to define an additional optional line with a HEADING in order to associate a name to the part.

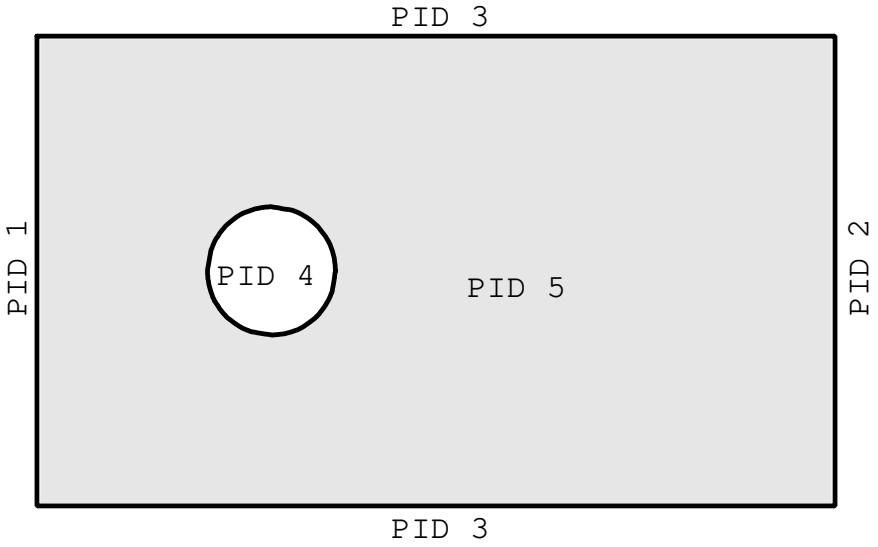
Title	1	2	3	4	5	6	7	8
Variable	HEADING							
Type	A							
Default	none							

Card 1	1	2	3	4	5	6	7	8
Variable	PID	SECID	MID					
Type	I	I	I					
Default	none	none	none					

Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card 2	1	2	3	4	5	6	7	8
Variable	SPID1	SPID2	SPID3	SPID4	SPID5	SPID6	SPID7	SPID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
PID	Part identifier for fluid volumes.
SECID	Section identifier defined by the *ICFD_SECTION card.
MID	Material identifier.
SPID1, …	Part IDs for the surface elements that define the volume mesh.




```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$   *ICFD_PART_VOL
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$ PART ID 5 is defined by the surfaces that enclose it.
$
*ICFD_PART_VOL
$
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$      pid       secid       mid
         5           1         1
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$      pid1      pid2      pid3      pid4      pid5      pid6      pid7      pid8
         1         2         3         4
```

*ICFD_SECTION

Purpose: Define a section for the incompressible flow solver.

Card 1	1	2	3	4	5	6	7	8
Variable	SID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
SID	Section identifier.

*ICFD_SET_NODE_LIST

Purpose: Only used in cases where the mesh is specified by the user (See *MESH_VOLUME_ELEMENT). Defines a set of nodes associated with a part ID on which boundary conditions can be applied.

Card 1	1	2	3	4	5	6	7	8
Variable	SID	PID						
Type	I	I						
Default	none	none						

Node List Card. Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card 2	1	2	3	4	5	6	7	8
Variable	NID1	NID2	NID3	NID4	NID5	NID6	NID7	NID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
SID	Set ID
PID	Associated Part ID.
NID1, …	Node IDs

Remarks:

1. The convention is the similar to the one used by the keyword *SET_NODE_LIST and serves a similar purpose.

*ICFD_SOLVER_SPLIT

Purpose: This keyword provides an option to trigger an iterative procedure on the fluid system. This procedure aims to bring more precision to the final pressure and velocity values but is often very time consuming. It must therefore be used with caution. It is intended only for special cases. For stability purposes, this method is automatically used for the first ICFD time step.

Card 1	1	2	3	4	5	6	7	8
Variable	NIT	TOL						
Type	I	F						
Default	1	10^{-3}						

VARIABLE	DESCRIPTION
NIT	Maximum Number of iterations of the system for each fluid time step. If TOL criteria is not reached after NIT iterations, the run will proceed.
TOL	Tolerance Criteria for the pressure residual during the fluid system solve.

*ICFD_SOLVER_TOL_FSI

Purpose: This keyword allows the user to change the default tolerance values for the Newton Raphson loop in the strong FSI analysis. *Care should be taken when deviating from the default values*.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT				
Type	F	F		I				
Default	10^{-5}	10^{-5}		1000				

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $\text{Residual}_{i+1} - \text{Residual}_i \leq \text{ATOL}$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(\text{Residual}_{i+1} - \text{Residual}_i)/\text{Residual}_{\text{initial}} \leq RTOL$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.

*ICFD_SOLVER_TOL_LSET

Purpose: This keyword allows the user to change the default tolerance values for the advection equation for levelset. *Care should be taken when deviating from the default values*.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT				
Type	F	F		I				
Default	10^{-8}	10^{-8}		1000				

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $\text{Residual}_{i+1} - \text{Residual}_i \leq \text{ATOL}$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(\text{Residual}_{i+1} - \text{Residual}_i)/\text{Residual}_{\text{initial}} \leq RTOL$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.

*ICFD_SOLVER_TOL_MMOV

Purpose: This keyword allows the user to change the default tolerance values for the mesh movement algorithm. *Care should be taken when deviating from the default values*.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT		DISPTOL		
Type	F	F		I		F		
Default	1e-8	1e-8		1000		0.		

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $\text{Residual}_{i+1} - \text{Residual}_i \leq \text{ATOL}$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(\text{Residual}_{i+1} - \text{Residual}_i)/\text{Residual}_{\text{initial}} \leq \text{RTOL}$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.
DISPTOL	Element deformation tolerance before a matrix reassembly is triggered. Default is 0. which means any element deformation detected will automatically trigger a matrix reassembly. Higher values will potentially save calculation times at the expense of accuracy.

*ICFD_SOLVER_TOL_MOM

Purpose: This keyword allows the user to change the default tolerance values for the momentum equation solve. *Care should be taken when deviating from the default values*.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT				
Type	F	F		I				
Default	10^{-8}	10^{-8}		1000				

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $\text{Residual}_{i+1} - \text{Residual}_i \leq \text{ATOL}$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(\text{Residual}_{i+1} - \text{Residual}_i)/\text{Residual}_{\text{initial}} \leq \text{RTOL}$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.

*ICFD_SOLVER_TOL_MONOLITHIC

Purpose: This keyword allows the user to change the default tolerance values for the monolithic solver. *Care should be taken when deviating from the default values*.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT				
Type	F	F		I				
Default	10^{-8}	10^{-8}		1000				

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $\text{Residual}_{i+1} - \text{Residual}_i \leq \text{ATOL}$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(\text{Residual}_{i+1} - \text{Residual}_i)/\text{Residual}_{\text{initial}} \leq \text{RTOL}$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.

*ICFD_SOLVER_TOL_PRE

Purpose: Change the default tolerance values for the Poisson equation for pressure. *Care should be taken when deviating from the default values*.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT	PREID	PTOL		
Type	F	F		I	I	F		
Default	10^{-8}	10^{-8}		1000	2	10^{-3}		

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $\text{Residual}_{i+1} - \text{Residual}_i \leq \text{ATOL}$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(\text{Residual}_{i+1} - \text{Residual}_i)/\text{Residual}_{\text{initial}} \leq \text{RTOL}$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.
PREID	Choice of Preconditioner for the Conjugate Gradient Solve: EQ.1: Diagonal Preconditioner EQ.2: Incomplete LU factorization EQ.5: Global MUMPS factorization
PTOL	Preconditioner tolerance (a.k.a Drop Tolerance if PREID = 2)

*ICFD_SOLVER_TOL_TEMP

Purpose: This keyword allows the user to change the default tolerance values for the heat equation. To be handled with great care.

Card 1	1	2	3	4	5	6	7	8
Variable	ATOL	RTOL		MAXIT				
Type	F	F		I				
Default	1e-8	1e-8		1000				

VARIABLE	DESCRIPTION
ATOL	Absolute convergence criteria. Convergence is achieved when $Residual_{i+1} - Residual_i \leq ATOL$. If a negative integer is entered, then that value will be used as a load curve ID for ATOL.
RTOL	Relative convergence criteria. Convergence is achieved when $(Residual_{i+1} - Residual_i)/Residual_{initial} \leq RTOL$. If a negative integer is entered, then that value will be used as a load curve ID for RTOL.
MAXIT	Maximum number of iterations allowed to achieve convergence. If a negative integer is entered, then that value will be used as a load curve ID for MAXIT.

*MESH

The keyword *MESH is used to create a mesh that will be used in the analysis. So far only tetrahedral (or triangular in 2-d) elements can be generated. The keyword cards in this section are defined in alphabetical order:

*MESH_BL

*MESH_BL_SYM

*MESH_EMBEDSHELL

*MESH_INTERF

*MESH_NODE

*MESH_SIZE_

*MESH_SIZE_SHAPE

*MESH_SURFACE_ELEMENT

*MESH_SURFACE_NODE

*MESH_VOLUME

*MESH_VOLUME_ELEMENT

*MESH_VOLUME_NODE

*MESH_VOLUME_PART

*MESH_BL

Purpose: Define a boundary-layer mesh as a refinement on the volume-mesh. The boundary layer mesh is constructed by subdividing elements near the surface.

Boundary Layer Cards. Define as many cards as are necessary. The next keyword ("*") card terminates this input.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	NELTH	BLTH	BLFE	BLST			
Type	I	I	F	F	I			
Default	none	none	0.	0.	0			

VARIABLE	DESCRIPTION
PID	Part identifier for the surface element
NELTH	Number of elements normal to the surface (in the boundary layer) is NELTH + 1. See Remark 5.
BLTH	Boundary layer mesh thickness if BLST = 1 or 2. Growth scale factor if BLST = 3. Ignored if BLST = 0. See Remark 5.
BLFE	Distance between the wall and the first volume mesh node if BLST = 3. Scaling coefficient if BLST = 1 or 2. Ignored if BLST =0. See Remark 5.
BLST	Boundary layer mesh generation strategy: EQ.0: $2^{NELTH+1}$ subdivisions based on surface mesh size (default). See Remark 1. EQ.1: Power law using BLTH and NELTH with BLFE as a scale factor. See Remark 2 and Figure 8-1. EQ.2: Geometric series based on BLTH and BLFE. See Remark 3 and Figure 8-2. EQ.3: Repartition following a growth scale factor (BLTH). See Remark 4 and Figure 8-3.

Remarks:

1. **BLST = 0.** For BLST = 0, for every additional NELTH, the automatic volume mesher will divide the elements closest to the surface by two so that the smallest element in the boundary layer mesh will have an aspect ratio of $2^{\text{NELTH}+1}$. A default boundary layer mesh thickness based on the surface mesh size will be chosen.

2. **BLST = 1.** For a constant repartition of the nodes in the boundary layer, use BLST = 1 with BLFE = 1. For BLST = 1, starting from the wall, the position of node n in the normal direction is given by:

$$X_n = \left(\frac{n}{\text{NELTH}+1}\right)^{[5\times(1-\text{BLFE})]} \frac{\text{BLTH}}{\sum_{i=1}^{\text{NELTH}+1}[i/(\text{NELTH}+1)]^{[5\times(1-\text{BLFE})]}}$$

3. **BLST = 2.** Setting BLFE = 1 makes BLST = 2 equivalent to BLST = 0 except that BLST = 2 allows you to specify the boundary layer thickness instead of automatically using the local surface mesh size. For BLST = 2, starting from BLTH from the wall, each newly inserted node will have its location closer to the wall, following this law:

$$X_n = (0.5 \times \text{BLFE})^n \times \text{BLTH}(1 - 0.5 \times \text{BLFE})$$

4. **BLST = 3.** For BLST = 3, starting from the wall, the position of node n in the normal direction is given by:

$$X_n = \sum_{i=0}^{n} \text{BLFE} \times \text{BLTH}^i \text{ with } 0 \le n \le \text{NELTH}$$

5. **Dynamically Changing Boundary Layer.** For NELTH, BLTH and BLFE, setting a negative value will point to a time dependent load curve which allows you to dynamically change the boundary layer settings in cases involving remeshing (including returning a 0 value for the number of elements which effectively disables the boundary layer).

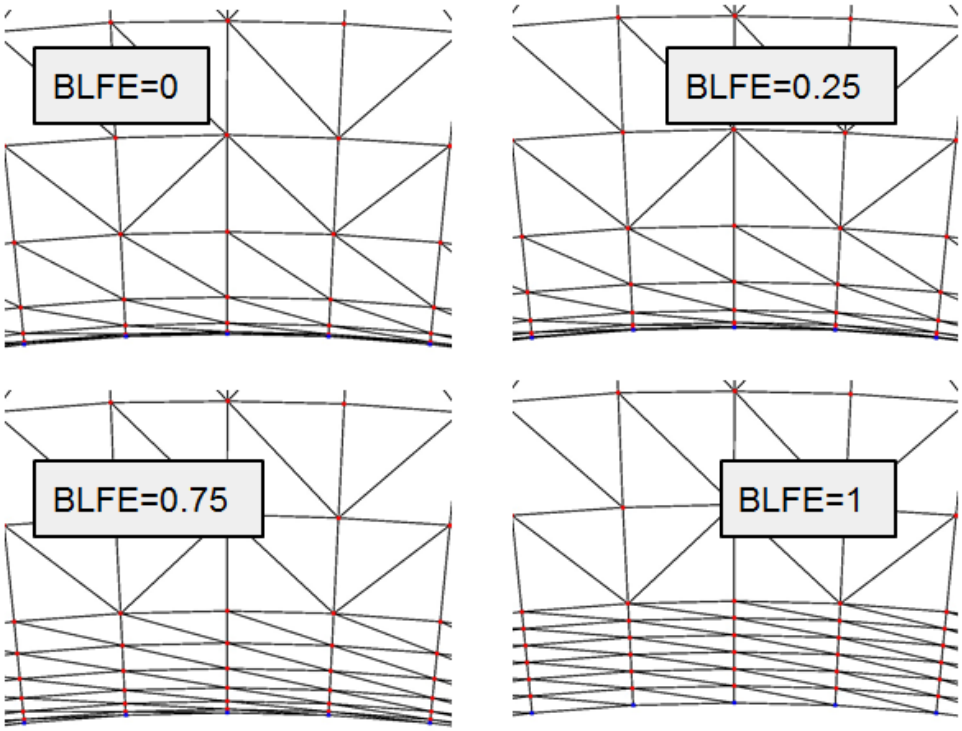


Figure 8-1. BLST = 1 example

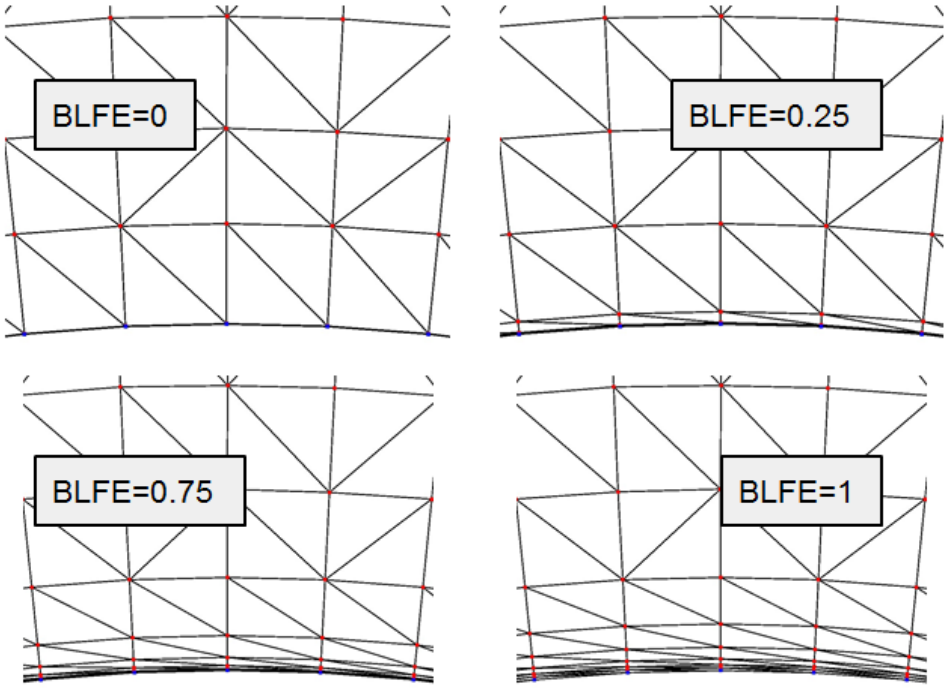


Figure 8-2. BLST = 2 example

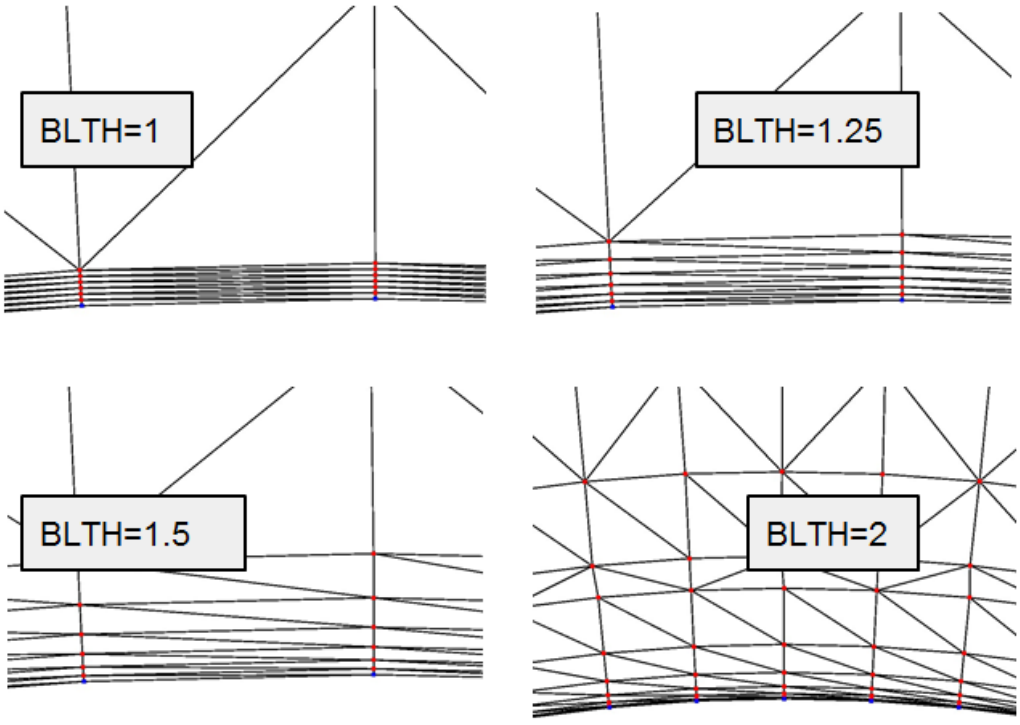


Figure 8-3. BLST = 3 example

*MESH_BL_SYM

Purpose: Specify the part IDs that will have symmetry conditions for the boundary layer. On these surfaces, the boundary layer mesh follows the surface tangent.

Boundary Layer with Symmetry Condition Cards. Define as many cards as necessary. The next "*" card terminates the input.

Card 1	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
PID1, …	Part identifiers for the surface element. This is the surface with symmetry.

*MESH_EMBEDSHELL

Purpose: Define surfaces that the mesher will embed inside the volume mesh. These surfaces will have no thickness and will conform to the rest of the volume mesh having matching nodes on the interface.

Card 1	1	2	3	4	5	6	7	8
Variable	VOLID							
Type	I							
Default	none							

Define as many cards as are necessary based on the number of PIDs (the next "*" card terminates the input.)

Card 2	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
VOLID	ID assigned to the new volume in the keyword *MESH_VOLUME. The surface mesh size will be applied to this volume.
PID*n*	Part IDs for the surface elements that will be embedded in the volume mesh.

*MESH_INTERF

Purpose: Define the surfaces that will be used by the mesher to specify fluid interfaces in multi-fluid simulations.

Card 1	1	2	3	4	5	6	7	8
Variable	VOLID							
Type	I							
Default	none							

Define as many cards as are necessary based on the number of PIDs. This input ends at the next keyword ("*") card.

Card 2	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
VOLID	ID assigned to the new volume in the keyword *MESH_VOLUME. The interface meshes will be applied to this volume.
PID*n*	Part IDs for the surface elements.

*MESH_NODE

Purpose: Define a fluid node and its coordinates. These nodes are used in the mesh generation process by the *MESH_SURFACE_ELEMENT keyword, or as user defined volume nodes by the *MESH_VOLUME_ELEMENT keyword.

Node Cards. Include one additional card for each node. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	NID	X		Y		Z				
Type	I	F		F		F				
Default	none	0		0		0				

VARIABLE	DESCRIPTION
NID	Node ID. A unique number with respect to the other surface nodes.
X	*x* coordinate.
Y	*y* coordinate.
Z	*z* coordinate.

Remarks:

1. The data card format for the *MESH_NODE keyword is identical to *NODE.
2. The *MESH_NODE keyword supersedes *MESH_SURFACE_NODE, which was for surfaces nodes as well as *MESH_VOLUME_NODE for, which was for volume nodes in user defined.

*MESH_SIZE

Purpose: Define the surfaces that will be used by the mesher to specify a local mesh size inside the volume. If no internal mesh is used to specify the size, the mesher will use a linear interpolation of the surface sizes that define the volume enclosure.

Card 1	1	2	3	4	5	6	7	8
Variable	VOLID							
Type	I							
Default	none							

Define as many cards as are necessary based on the number of PIDs (the next "*" card terminates the input.).

Card 2	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
VOLID	ID assigned to the new volume in the keyword *MESH_VOLUME. The mesh sizing will be applied to this volume.
PID*n*	Part IDs for the surface elements that are used to define the mesh size next to the surface mesh.

*MESH_SIZE_SHAPE

Purpose: Defines a local mesh size in specific zones corresponding to given geometrical shapes (box, sphere, cylinder and polynomial). The solver will automatically apply the conditions specified during the generation of the volume mesh. This zone does not need to be entirely defined in the volume mesh. In the polynomial case, it is recommended to define several zones for a better mesh size control.

Remeshing Control Card sets:

Add as many *remeshing control cards* paired with a *case card* as desired. The input of such pairs ends at the next keyword "*" card.

Remeshing Control. First card specifies whether to maintain this mesh sizing criterion through a remesh operation.

Card 1	1	2	3	4	5	6	7	8
Variable	SNAME	FORCE	METHOD	BT	DT			
Type	A	I	I	F	F			
Default	none	0	0	0.	1.E12			

Box Case. Card 2 for SNAME = "box" and METHOD = 0

Cards 2	1	2	3	4	5	6	7	8
Variable	MSIZE	PMINX	PMINY	PMINZ	PMAXX	PMAXY	PMAXZ	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

Sphere Case. Card 2 for SNAME = "sphere" and METHOD = 0

Cards 2	1	2	3	4	5	6	7	8
Variable	MSIZE	RADIUS	CENTERX	CENTERY	CENTERZ			
Type	F	F	F	F	F			
Default	none	none	none	none	none			

Cylinder Case. Card 2 for SNAME = "cylinder" and METHOD = 0

Cards 2	1	2	3	4	5	6	7	8
Variable	MSIZE	RADIUS	PMINX	PMINY	PMINZ	PMAXX	PMAXY	PMAXZ
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Polynomial Case. Card 2 for SNAME = "pol" and METHOD = 0

Cards 2	1	2	3	4	5	6	7	8
Variable	MSIZE	X	Y	Z	NX	NY	NZ	
Type	F	F	F	F	F	F	F	
Default	none	none	none	none	none	none	none	

Card 2 for METHOD = 1

Cards 2	1	2	3	4	5	6	7	8
Variable	MSIZE	RADIUS	PTID1	PTID2				
Type	F	F	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
SNAME	Shape name. Possibilities include "box", "cylinder", "pol" and "sphere"
FORCE	Force to keep the mesh size criteria even after a remeshing is done. EQ.0: Off, mesh size shape will be lost if a remeshing occurs EQ.1: On.
METHOD	Specifies which method to use when defining the second card. EQ.0: Default, directly input the coordinates. EQ.1: Define the coordinates via the introduction of ICFD_DEFINE_POINT IDs. The biggest advantage of using this method is that the ICFD_DEFINE_POINTs are allowed to move which allows the user to control how the mesh size area should evolve function of time in cases where there is remeshing.
BT/DT	Birth and death time of the mesh size area in cases where remeshing occurs.
MSIZE	Mesh size that needs to be applied in the zone of the shape defined by SNAME
PMIN[X, Y, Z]	x, y, or z value for the point of minimum coordinates
PMAX[X, Y, Z]	x, y, or z value for the point of maximum coordinates
CENTER[X, Y, Z]	Coordinates of the sphere center in cases where SNAME is sphere
RADIUS	Radius of the sphere if SNAME is Sphere or of the cross section disk if SNAME is Cylinder.

VARIABLE	DESCRIPTION
X/Y/Z	Coordinates of starting point in cases where SNAME is pol.
NX/NY/NZ	Direction in which mesh size will be applied in cases where SNAME is pol.
PTID1	Point ID1 referring to ICFD_DEFINE_POINT. Replaces PMIN, X/Y/Z or CENTER for the various SNAME cases.
PTID2	Point ID2. Not needed if SNAME is Sphere. Replaces PMAX or NX/NY/NZ for the various SNAME cases.

*MESH_SURFACE_ELEMENT

Purpose: Specify a set of surface elements (quadrilateral or triangular in 3D and linear segments in 2D) that will be used by the mesher to construct a volume mesh. These surface elements may define the enclosed volume to be meshed, or alternatively they can be used to specify different mesh sizes inside the volume (see card *MESH_SIZE**).**

Each solver that supports the *MESH volume mesher for building its volume mesh uses the PID given for each surface element specified with this keyword differently.

1. For the *ICFD solver, the *ICFD_PART keyword references the surface mesh PIDs.

2. For the *DUALCESE solver, the MSPIDs in *DUALCESE_BOUNDARY_... keywords cards reference the surface element PIDs. Note that when the dual CESE solver uses *MESH_SURFACE_ELEMENT, this card defines which boundary faces belong to each mesh surface PID. In this case, no other mechanism exists for defining these PIDs. Also, when the dual CESE solver is defined with *MESH cards, there should not be any *DUALCESE_SEGMENTSET cards related to the dual CESE mesh.

Surface Element Card. Define as many cards as necessary. The next keyword ("*") card terminates this input.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4				
Type	I	I	I	I	I	I				
Default	none	none	none	none	none	none				

VARIABLE	DESCRIPTION
EID	Element ID. A unique number with respect to all *MESH_SURFACE_ELEMENTS cards.
PID	Mesh surface part ID. A unique identifier for the surface to which this mesh surface element belongs.
N1	Nodal point 1
N2	Nodal point 2

VARIABLE	DESCRIPTION
N3	Nodal point 3
N4	Nodal point 4

Remarks:

1. **Defining Surface Elements.** The convention for defining surface elements is the same as for *ELEMENT_SHELL. In the case of a triangular face, N3 = N4. In 2D N2 = N3 = N4. Note that the accepted card format is 6i8 (not 6i10).

*MESH_SURFACE_NODE

Purpose: Define a node and its coordinates. These nodes will be used in the mesh generation process by the *MESH_SURFACE_ELEMENT keyword.

*MESH_NODE supersedes this card; so please use *MESH_NODE instead of this card.

Surface Node Cards. Include one card for each node. Include as many cards a necessary. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	NID	X		Y		Z				
Type	I	F		F		F				
Default	none	0		0		0				

VARIABLE	DESCRIPTION
NID	Node ID. This NID must be unique within the set of surface nodes.
X	*x* coordinate.
Y	*y* coordinate.
Z	*z* coordinate.

*MESH_VOLUME

Purpose: This keyword defines the volume space that will be meshed. The boundaries of the volume are the surfaces defined by *MESH_SURFACE_ELEMENT. The surfaces listed have to be non-overlapping, and should not leave any gaps or open spaces between the surface boundaries. On the boundary between two neighbor surfaces, nodes have to be in common (no duplicate nodes) and should match exactly on the interface. They are defined by the keyword *MESH_SURFACE_NODE. This card will be ignored if the volume mesh is specified by the user and not generated automatically.

Card 1	1	2	3	4	5	6	7	8
Variable	VOLID							
Type	I							
Default	none							

Define as many cards as are necessary based on the number of PIDs (the next "*" card terminates the input.)

Card 2	1	2	3	4	5	6	7	8
Variable	PID1	PID2	PID3	PID4	PID5	PID6	PID7	PID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
VOLID	ID assigned to the new volume.
PID*n*	Part IDs for the surface elements that are used to define the volume.

*MESH_VOLUME_ELEMENT

Purpose: Specify a set of volume elements for the fluid volume mesh in cases where the volume mesh is specified by the user and not generated automatically. The nodal point are specified in the *MESH_VOLUME_NODE keyword. Only tetrahedral elements are supported (triangles in 2D).

Volume Element Card. Define as many cards as necessary. The next "*" card terminates the input.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4				
Type	I	I	I	I	I	I				
Default	none	none	none	none	none	none				

VARIABLE	DESCRIPTION
EID	Element ID. A unique number with respect to all *MESH_VOLUME_ELEMENTS cards.
PID	Part ID. A unique part identification number.
N1	Nodal point 1.
N2	Nodal point 2.
N3	Nodal point 3.
N4	Nodal point 4.

Remarks:

1. The convention is the same used by the keyword *ELEMENT_SOLID.

*MESH_VOLUME_NODE

Purpose: Define a node and its coordinates. This keyword is only used in cases where the fluid volume mesh is provided by the user and is not automatically generated. It serves the same purpose as the *NODE keyword for solid mechanics. Only tetrahedral elements are supported.

*MESH_NODE supersedes this card; so please use *MESH_NODE instead of this card.

Volume Node Cards. Include as many cards in the following format as desired. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	NID	X		Y		Z				
Type	I	F		F		F				
Default	none	0		0		0				

VARIABLE	DESCRIPTION
NID	Node ID. A unique number with respect to the other volume nodes.
X	*x* coordinate.
Y	*y* coordinate.
Z	*z* coordinate.

*MESH_VOLUME_PART

Purpose: Associate a volume part number created by a *MESH_VOLUME card with the part number of a part card from a selected solver (designated by the SOLVER field).

Mesh Volume Part Card. Include as many cards in the following format as desired. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	VOLPRT	SOLPRT	SOLVER					
Type	I	I	A					
Default								

VARIABLE	DESCRIPTION
VOLPRT	Part ID of a volume part created by a *MESH_VOLUME card.
SOLPRT	Part ID of a part created using SOLVER's part card.
SOLVER	Name of a solver using a mesh created with *MESH cards.

*STOCHASTIC

The keyword *STOCHASTIC is used to describe the particles and numerical details for solving a set of stochastic PDEs. Currently, there are two types of stochastic PDE models in the code: a spray model and a model of embedded particles in TBX explosives. The keyword cards for using these models are:

*STOCHASTIC_SPRAY_PARTICLES

*STOCHASTIC_TBX_PARTICLES

An additional option "_TITLE" may be appended to all *STOCHASTIC keywords. If this option is used, then an 80 character string is read as a title from the first card of that keyword's input. At present, LS-DYNA does not make use of the title. Inclusion of titles gives greater clarity to input decks.

*STOCHASTIC_SPRAY_PARTICLES

Purpose: Specify particle and other model details for spray modeling using stochastic PDEs that approximate such processes. A pair of cards is required to specify the characteristics of each nozzle (cards 3 and 4 describe the first nozzle).

Card 1	1	2	3	4	5	6	7	8
Variable	INJDIST	IBRKUP	ICOLLDE	IEVAP	IPULSE	LIMPR	IDFUEL	
Type	I	I	I	I	I	I	I	
Default	1	none	none	0	none	none	1	

Card 2	1	2	3	4	5	6	7	8
Variable	RHOP	TIP	PMASS	PRTRTE	STRINJ	DURINJ		
Type	F	F	F	F	F	F		

Nozzle card 1: Provide as many pairs of nozzle cards 1 and 2 as necessary. This input ends at the next keyword ("*") card (following a nozzle card 2).

Card 3	1	2	3	4	5	6	7	8
Variable	XORIG	YORIG	ZORIG	SMR	VELINJ	DRNOZ	DTHNOZ	
Type	F	F	F	F	F	F	F	

Nozzle card 2: Provide as many pairs of nozzle cards 1 and 2 as necessary. This input ends at the next keyword ("*") card.

Card 4	1	2	3	4	5	6	7	8
Variable	TILTXY	TILTXZ	CONE	DCONE	ANOZ	AMP0		
Type	F	F	F	F	F	F		

VARIABLE	DESCRIPTION
INJDIST	Spray particle size distribution: EQ.1: uniform EQ.2: Rosin-Rammler (default) EQ.3: Chi-squared degree of 2 EQ.4: Chi-squared degree of 6
IBRKUP	Type of particle breakup model: EQ.0: off (no breakup) EQ.1: TAB EQ.2: KHRT
ICOLLDE	Turn collision modeling on or off
IEVAP	Evaporation flag: EQ.0: off (no evaporation) EQ.1: Turn evaporation on (see Remark 1)
IPULSE	Type of injection: EQ.0: continuous injection EQ.1: sine wave EQ.2: square wave
LIMPRT	Upper limit on the number of parent particles modeled in this spray. This is not used with the continuous injection case (IPULSE = 0).

VARIABLE	DESCRIPTION
IDFUEL	Selected spray liquid fuels: EQ.1: (Default), H_2O EQ.2: Benzene, C_6H_6 EQ.3: Diesel # 2, $C_{12}H_{26}$ EQ.4: Diesel # 2, $C_{13}H_{13}$ EQ.5: Ethanol, C_2H_5OH EQ.6: Gasoline, C_8H_{18} EQ.7: Jet-A, $C_{12}H_{23}$ EQ.8: Kerosene, $C_{12}H_{23}$ EQ.9: Methanol, CH_3OH EQ.10: N-dodecane, $C_{12}H_{26}$
RHOP	Particle density
TIP	Initial particle temperature.
PMASS	Total particle mass
PRTRTE	Number of particles injected per second for continuous injection.
STRINJ	Start of injection(s)
DURINJ	Duration of injection(s)
XORIG	X-coordinate of center of a nozzle's exit plane
YORIG	Y-coordinate of center of a nozzle's exit plane
ZORIG	Z-coordinate of center of a nozzle's exit plane
SMR	Sauter mean radius
VELINJ	Injection velocity
DRNOZ	Nozzle radius
DTHNOZ	Azimuthal angle (in degrees measured counterclockwise) of the injector nozzle from the j = 1 plane.

VARIABLE	DESCRIPTION
TILTXY	Rotation angle (in degrees) of the injector in the x-y plane, where 0.0 points towards the 3 o'clock position (j = 1 line), and the angle increases counterclockwise from there.
TILTXZ	Inclination angle (in degrees) of the injection in the x-z plane, where 0.0 points straight down, x > 0.0 points in the positive x direction, and x < 0.0 points in the negative x direction.
CONE	Spray mean cone angle (in degrees) for hollow cone spray; spray cone angle (in degrees) for solid cone spray.
DCONE	Injection liquid jet thickness in degrees.
ANOZ	Area of injector
AMP0	Initial amplitude of droplet oscillation at injector

Remarks:

1. When IEVAP = 1, the keyword input file must be modified in a fashion similar to a chemistry problem. This is illustrated in a portion of an example keyword file below. That is, the following keywords need to be used, along with the inclusion of other chemistry-related files (i.e. evap.inp and the corresponding thermodynamics data file):

   *CHEMISTRY_MODEL

   *CHEMISTRY_COMPOSITION

   *CHEMISTRY_CONTROL_FULL

   *CESE_INITIAL_CHEMISTRY

```
$ Setup stochastic particles
$
*STOCHASTIC_SPRAY_PARTICLES
$  injdist    ibrkup  icollide     ievap    ipulse    limprt    fuelid
         3         1         0         1         0    100000         1
$     rhop       tip pmass[Kg]    prtrte   str_inj   dur_inj
    1000.0      300.      0.01     1.0e7       0.0      10.0
$  the next card is needed for fireball position and max.  particle velocity:
$    XORIG     YORIG     ZORIG       SMR   Velinj      Drnoz    Dthnoz
     0.005     0.005    1.0e-5    5.0e-6    200.0     9.0e-5
$   TILTXY    TILTXZ      CONE     DCONE      ANOZ       AMP0
       0.0       0.0      15.0      15.0   2.5e-8        0.0
$
*CHEMISTRY_MODEL
$ model_id    jacsel    errlim
```

```
        10         1       0.0
  evap.inp
 therm.dat
  tran.dat
$
*CHEMISTRY_COMPOSITION
$  comp_id  model_id
        11        10
$  molefra   Species
       1.0        O2
      3.76        N2
$
*CHEMISTRY_CONTROL_FULL
$   sol_id    errlim
         5
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$ Set global initial conditions for fluid
$
*CESE_INITIAL_CHEMISTRY
$   sol_id   comp_id
         5        11
$INITIAL CONDITIONS
$      uic       vic       wic       ric       pic       tic       hic
       0.0       0.0       0.0       1.2   101325.     300.0       0.0
```

*STOCHASTIC_TBX_PARTICLES

Purpose: Specify particle and other model details for stochastic PDEs that model embedded particles in TBX explosives. Note that the components listed on the corresponding *CHEMISTRY_COMPOSITION card are in terms of molar concentrations of the species (in units of moles/[length]^3, where "[length]" is the user's length unit).

For further information on the theory of the TBX model that has been implemented, a document on this topic can be found at this URL:

http://www.lstc.com/applications/cese_cfd/documentation

Card 1	1	2	3	4	5	6	7	8
Variable	PCOMB	NPRTCL	MXCNT	PMASS	SMR	RHOP	TICP	T_IGNIT
Type	I	I	I	F	F	F	F	F
Default	0	none	none	none	none	none	none	none

Card 2	1	2	3	4	5	6	7	8
Variable	INITDST	AZIMTH	ALTITD	CPS/CVS	HVAP	EMISS	BOLTZ	
Type	I	F	F	F	F	F	F	
Default	1	none	none	none	none	none	none	
Remarks						1	1	

Card 3	1	2	3	4	5	6	7	8
Variable	XORIG	YORIG	ZORIG	XVEL	YVEL	ZVEL	FRADIUS	
Type	F	F	F	F	F	F	F	
Default	none	none	none	0.0	0.0	0.0	none	

VARIABLE	DESCRIPTION
PCOMB	Particle combustion model EQ.0: no burning EQ.1: K-model
NPRTCL	Initial total number of parent particles (discrete particles for calculation)
MXCNT	Maximum allowed number of parent particles (during the simulation)
PMASS	Total particle mass
SMR	Sort mean particle radius
RHOP	Particle density
TICP	Initial particle temperature
T_IGNIT	Particle ignition temperature
INITDST	Initial particle distribution EQ.1: spatially uniform EQ.2: Rosin-Rammler EQ.3: Chi-squared
AZIMTH	Angle in degrees from x-axis in x-y plane of reference frame of TBX explosive (0 < AZMITH < 360)
ALTITD	Angle in degrees from z-axis of reference frame of TBX explosive (0 < ALTITD < 180)
CPS/CVS	Heat coefficient
HVAP	Latent heat of vaporization
EMISS	Particle emissivity
BOLTZ	Boltzmann coefficient
XORIG	x-coordinate of the origin of the initial reference frame of the TBX explosive

VARIABLE	DESCRIPTION
YORIG	*y*-coordinate of the origin of the initial reference frame of the TBX explosive
ZORIG	*z*-coordinate of the origin of the initial reference frame of the TBX explosive
XVEL	*x*-component of the initial particle velocity the TBX explosive
YVEL	*y*-component of the initial particle velocity the TBX explosive
ZVEL	*z*-component of the initial particle velocity the TBX explosive
FRADIUS	Radius of the explosive area.

Remarks:

1. If radiation heat transfer is being modeled, then EMISS and BOLTZ are required.

*LSO

These cards provide a general data output mechanism, causing the creation of a sequence of LSDA files. This facility is intended to allow several different time sequences of data to be output in the same simulation. In addition, any number of domains (and any number of variables on those domains) may be specified within each time sequence. The keyword cards in this section are defined in alphabetical order:

*LSO_DOMAIN

*LSO_ID_SET (*not available in the single-precision version of LS-DYNA*)

*LSO_POINT_SET

*LSO_TIME_SEQUENCE

*LSO_VARIABLE_GROUP

Note that only the mechanics solver is available in the single-precision version of LS-DYNA, and therefore, only LSO mechanics variables are available for output from single precision LS-DYNA. These mechanics variables are listed by domain type in a separate document. This document (LSO_VARIABLES.TXT) is created by running the command: LS-DYNA print_lso_doc. Contrary to LSO_VARIABLES.TXT, element quantities such as stress are not available for output from the mechanics solver to the "lso" database.

An additional option "_TITLE" may be appended to all *LSO keywords. If this option is used, then an 80 character string is read as a title from the first card of that keyword's input. At present, LS-DYNA does not make use of the title. Inclusion of titles gives greater clarity to input decks.

*LSO_DOMAIN

Purpose: This command provides a way to specify variables on a subset of the domain for a given solver. This domain can be a subset of the mesh used by that solver, a set of output points created with *LSO_POINT_SET, or a set of objects created with *LSO_ID_SET. The frequency and duration of the output for any given domain is determined by each *LSO_TIME_SEQUENCE card that references this *LSO_DOMAIN card. Note that for the single-precision version of LS-DYNA, the only allowed value of SOLVER_NAME = MECH.

Card 1	1	2	3	4	5	6	7	8
Variable	DOMAIN_TYPE							
Type	A							

Card 2	1	2	3	4	5	6	7	8
Variable	SOLVER_NAME							
Type	A							

Special Domains Card. Card 3 when DOMAIN_TYPE is one of ROGO, CIRCUIT, THIST_POINT or TRACER_POINT.

Card 3	1	2	3	4	5	6	7	8
Variable	OUTID	REFID		REDUCT				
Type	I	I		I				
Default	none	none		none				

Miscellaneous Domain Card. Card 3 when DOMAIN_TYPE is one of NODE, PART, SEGMENT, SURFACE_NODE, SURFACE_ELEMENT, VOLUME_ELEMENT, SURFACE_PART, VOLUME_PART.

Card 3	1	2	3	4	5	6	7	8
Variable	OUTID	REFID	OVERRIDE	REDUCT				
Type	I	I	I	I				
Default	none	0	0	none				

Variable Name Card. Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card 4	1	2	3	4	5	6	7	8
Variable	VARIABLE_NAME							
Type	A							

VARIABLE	DESCRIPTION
DOMAIN_TYPE	The type of domain for which LSO output may be generated.
SOLVER_NAME	Selects the solver from which data is output on this domain. Accepted entries so far are "MECH", "EM", "CESE", and "ICFD".
OUTID	LSO domain ID associated with this domain, and used by *LSO_TIME_SEQUENCE cards.
REFID	Support set ID. This can be a set defined by a *SET card, a *LSO_ID_SET, card, or a *LSO_POINT_SET card. Unless OVERRIDE is specified, this set must be of the same type as DOMAIN_TYPE.
OVERRIDE	If non-zero, then REFID is interpreted as: EQ.1: a PART set for SOLVER_NAME EQ.2: a PART set of volume parts created with a *LSO_ID_SET card (volume parts are defined with *MESH_VOLUME cards).

VARIABLE	DESCRIPTION
	EQ.3: a PART set of surface parts created with a *LSO_ID_SET card (surface parts are defined with *MESH_SURFACE_ELEMENT cards).
	EQ.4: a set of segment sets created with a *LSO_ID_SET card.
REDUCT	A function that operates on the entire domain and returns a single value for scalar variables, three values for vector variables, or 6 values for symmetric tensor variables. For REDUCT="range", the number of returned values doubles. The following are the supported functions: EQ.BLANK: no reduction (default) EQ."none": Same as above EQ."avg": the average by component EQ."average": Same as above EQ."min": the minimum by component EQ."minimum": Same as above EQ."max": the maximum by component EQ."maximum": Same as above EQ."sum": the sum by component EQ."range": the minimum by component followed by the maximum by component
VARIABLE_NAME	Either the name of a single output variable or a variable group. See remarks.

	EM
VECTORS	magneticF
	electricFie
	vecpotFie
	currentDens
SCALARS	ScalarPoter

Remarks:

1. Supported choices for VARIABLE_NAME are listed by DOMAIN_TYPE for each SOLVER_NAME in a separate document. This document (LSO_VARIABLES.TXT) is created by running the command: LS-DYNA print_lso_doc. The following table shows a sample of the point output variables available when DOMAIN_TYPE = THIST_POINT:

*LSO_ID_SET

Purpose: Provides a way to create a set of existing sets (segment sets), or to define a set that is not available with other set-related keyword cards. These are then used in other *LSO cards to specify LSO output. This card is not available in the single precision version of LS-DYNA.

Card 1	1	2	3	4	5	6	7	8
Variable	SETID	TYPE	SOLVER					
Type	I	A	A					
Default	none	none	MECH					

Referenced IDs. Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card	1	2	3	4	5	6	7	8
Variable	ID1	ID2	ID3	ID4	ID5	ID6	ID7	ID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
SETID	Identifier for this ID set.

VARIABLE	DESCRIPTION
TYPE	The kind of IDs in this set: EQ.'SEG_SETS': Each ID is a segment set connected with SOLVER. EQ.'CIRCUIT': Each ID is a circuit ID (from *EM cards) EQ.'SURF_PARTS': Each ID is a surface part number (See *MESH_SURFACE_ELEMENT) EQ.'VOL_PARTS': Each ID is a volume part number (See *MESH_VOLUME) EQ.'SURF_ELES': Each ID is a surface element number (See *MESH_SURFACE_ELEMENT)
SOLVER	Name of the solver (MECH, ICFD, CESE, EM, …)
ID1, …	IDs of the TYPE kind.

*LSO_POINT_SET

Purpose: Define a list of points used to sample variables in time. Of the different sampling methods, the most common one is to specify points for time history output.

Card 1	1	2	3	4	5	6	7	8
Variable	SETID	USE						
Type	I	I						
Default	none	1						
Remarks		1						

Point Cards. Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card 2	1	2	3	4	5	6	7	8
Variable	X	Y	Z					
Type	F	F	F					
Default	none	none	none					

VARIABLE	DESCRIPTION
SETID	Identifier for this point set which is used by *LSO_DOMAIN
USE	Points in this set are used as: EQ.1: fixed time history points (default) EQ.2: positions of tracer particles
X, Y, Z	Coordinates of a point. As many points as desired can be specified.

Remarks:

1. **USE.** For USE = 1, with the ICFD and CESE solvers, the fixed points must remain inside the fluid mesh or a zero result is returned, while for the EM solver, the points can be defined inside the conductors or in the air. In the latter case, the fields will be computed using a Biot-Savart type integration. For USE = 2, a massless tracer particle is tracked for the ICFD and CESE solvers using their local velocity field to integrate the position of each particle in time.

*LSO_TIME_SEQUENCE

Purpose: This command provides users with maximum flexibility in specifying exactly what they want to have appear in the output LSO binary database. Each instance of the *LSO_TIME_SEQUENCE command creates a new time sequence with an independent output frequency and duration. Furthermore, while the default domain for each output variable will be the entire mesh on which that variable is defined, at all selected snapshot times, the *LSO_DOMAIN keyword commands can be used to specify that output will only occur on a portion of SOLVER_NAME's mesh, and for a limited time interval, or that it will occur at a set of points (see *LSO_POINT_SET), or over a set of object IDs (see *LSO_ID_SET). Note that for the single-precision version of LS-DYNA, the only allowed value of SOLVER_NAME = MECH.

Card 1	1	2	3	4	5	6	7	8
Variable	SOLVER_NAME							
Type	A							

Card 2	1	2	3	4	5	6	7	8
Variable	DT	LCDT	LCOPT	NPLTC	TBEG	TEND		
Type	F	I	I	I	F	F		
Default	0.0	0	1	0	0.0	0.0		
Remarks	1	1	1	1				

Domain IDs. Provide as many cards as necessary. This input ends at the next keyword ("*") card, or when a global variable name card appears

Card 3	1	2	3	4	5	6	7	8
Variable	DOMID1	DOMID2	DOMID3	DOMID4	DOMID5	DOMID6	DOMID7	DOMID8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

Global variable names. Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card 4	1	2	3	4	5	6	7	8
Variable	GLOBAL_VAR							
Type	A							

VARIABLE	DESCRIPTION
SOLVER_NAME	Selects the solver from which data is output in this time sequence. Accepted entries so far are 'MECH', 'EM', 'CESE' and 'ICFD'
DT	Time interval between outputs.
LCDT	Optional load curve ID specifying the time interval between dumps.
LCOPT	Flag to govern behavior of plot frequency load curve: EQ.1: At the time each plot is generated, the load curve value is added to the current time to determine the next plot time (this is the default behavior). EQ.2: At the time each plot is generated, the next plot time T is computed so that T = the current time plus the load curve value at the time T. EQ.3: A plot is generated for each ordinate point in the load curve definition. The actual value of the load curve is ignored.

VARIABLE	DESCRIPTION
NPLTC	DT = ENDTIM/NPLTC overrides DT specified in the first field.
TBEG	The problem time at which to begin writing output to this time sequence
TEND	The problem time at which to terminate writing output to this time sequence
DOMID1, …	Output set ID defining the domain over which variable output is to be performed in this time sequence. Each DOMID refers to the domain identifier in an *LSO_DOMAIN keyword card.
GLOBAL_VAR	The name of a global output variable computed by SOLVER_NAME. This variable must have a single value (scalar, vector, or tensor), and therefore does not depend upon any DOMID. Any number of such variables may be specified with a given time sequence. These variables are listed as having "global" domain for SOLVER_NAME in a separate document. This document (LSO_VARIABLES.TXT) is created by running the command: LS-DYNA print_lso_doc.

Remarks:

1. If LCDT is nonzero, then it is used and DT and NPLTC are ignored. If LCDT is zero and NPLTC is non-zero, then NPLTC determines the snapshot time increment. If LCDT and NPLTC are both zero, then the minimum non-zero time increment specified by DT is used to determine the snapshot times.

*LSO_VARIABLE_GROUP

Purpose: To provide a means of defining a shorthand name for a group of variables. That is, wherever the given group name is used, it is replaced by the list of variables given in this command. Note that for the single-precision version of LS-DYNA, the only allowed value of SOLVER_NAME = MECH.

Card 1	1	2	3	4	5	6	7	8
Variable	SOLVER_NAME							
Type	A							

Card 2	1	2	3	4	5	6	7	8
Variable	DOMAIN_TYPE							
Type	A							

Card 3	1	2	3	4	5	6	7	8
Variable	GROUP_NAME							
Type	A							

List Of Variables In Group. Provide as many cards as necessary. This input ends at the next keyword ("*") card

Card 4	1	2	3	4	5	6	7	8
Variable	VAR_NAME							
Type	A							

VARIABLE	DESCRIPTION
SOLVER_NAME	Selects the solver for which data is output in a time sequence.

VARIABLE	DESCRIPTION
DOMAIN_TYPE	Name of the type of domain on which each VAR_NAME is defined.
GROUP_NAME	Name of (or alias for) the group of names given by the listed VAR_NAMEs
VAR_NAME	The name of an output variable computed by SOLVER_NAME

Remarks:

1. Valid VAR_NAMEs depend both upon the SOLVER_NAME and the DOMAIN_TYPE. These variables are listed by DOMAIN_TYPE for each SOLVER_NAME in a separate document. This document (LSO_VARIABLES.TXT) is created by running the command: LS-DYNA print_lso_doc.