



AD FALCON API Manual

General Input File Structure for FALCON

Javad Ghorbani

March 26, 2026

Contents

- 1 General Input File Structure for FALCON 4**
- 1.1 Syntax 4
 - 1.1.1 Numeric Literals (Numbers) 4
 - 1.1.2 Lists and Ranges (IDs, Steps, Nodes, Elements) 4
 - 1.1.3 Booleans 5
 - 1.1.4 Surrogate-only value tags (\$name) 5
 - 1.1.5 Whitespace & Case Rules (What Is Case-Insensitive vs Case-Sensitive) 5
 - 1.1.6 Whitespace Normalization (How Spaces Are Dealt With) 5
 - 1.1.7 Comments 5
 - 1.1.8 Section Markers 6
 - 1.1.9 Include (Modular Files) 6
 - 1.1.10 Path Resolution (Working / Project Directory) 6
 - 1.1.11 Property Markers 7
 - 1.1.12 Multiple Definitions (Repeating Lines and Blocks) 7
- 1.2 Core Required Sections 8
 - 1.2.1 Checkpoint / restart (optional) 8
- 1.3 Optional Sections (can appear in many orders) 8
 - 1.3.1 Constraints & modifiers 8
 - 1.3.2 Loading & Forces 9
 - 1.3.3 Wave Propagation & Mesh Control 9
 - 1.3.4 Advanced Features 9
 - 1.3.5 Initialization 12
 - 1.3.6 Output & Monitoring 13
 - 1.3.7 Parametric Studies & Uncertainty Quantification 13
 - 1.3.8 Modular Files 13
- 1.4 Summary of Input Sections 13
- 1.5 Analysis Type 19
- 1.6 Nodes 20
- 1.7 Materials 20
- 1.8 Elements 21
- 1.9 Initial Assignments 21
- 1.10 Step Definitions 22
- 1.11 Additional Sections Reference 22
 - 1.11.1 Boundary Conditions & Constraints 22
 - 1.11.2 Loading & Forces 22
 - 1.11.3 Wave Propagation & Mesh Control 23
 - 1.11.4 Advanced Features 23
 - 1.11.5 Output & Monitoring 23
 - 1.11.6 Parametric Studies & Uncertainty Quantification 23

1.11.7 Modular Files 23

1.12 Error Codes and Troubleshooting 24



1 General Input File Structure for FALCON

This document outlines the complete structure and syntax for sections in a FALCON input file, based on the actual input reader implementation. Sections can appear in flexible order, but the following sequence is recommended for logical workflow, and for the **core sections the order of appearance is mandatory**.

1.1 Syntax

FALCON is designed to be tolerant to common formatting variations. In general:

- Leading/trailing whitespace is ignored.
- Tabs and multiple spaces are treated as separators.
- Section headers (% ...) and directive keys (@...) are parsed **case-insensitively**.
- Around : in directive headers, whitespace is optional (@Key:Value, @Key: Value, @Key : Value all work).

1.1.1 Numeric Literals (Numbers)

FALCON accepts standard decimal and scientific-notation numbers (finite values only).

Format	Examples
Integer	10, -3, +7
Decimal	10.0, 5., .25, -0.01
Scientific	1e1, 1E1, 1e+1, 1.0E+1, -2.5e-3

Notes: - Many integer fields (IDs, step numbers, frequencies) also accept *integer-equivalent* scientific/decimal forms such as 1e1 or 10.0.

1.1.2 Lists and Ranges (IDs, Steps, Nodes, Elements)

In list contexts, IDs and step indices can typically be written using:

Feature	Examples	Meaning
Separators	1 2 3, 1,2,3, 1;2;3, 1, 2; 3	Mixing whitespace/comma/semicolon is allowed
Inclusive ranges	1-5, 10:12	Expands to 1 2 3 4 5 or 10 11 12

Notes: - For ranges, avoid spaces around the range operator (1-5, not 1 - 5). - Some lists also accept the special keyword ALL (section-specific).

1.1.3 Booleans

Boolean values are parsed case-insensitively:

Meaning	Accepted values
True	Yes, Y, True, T, On, 1
False	No, N, False, F, Off, 0

1.1.4 Surrogate-only value tags (\$name)

You may tag any numeric value by placing a \$name token immediately before the number. These tags are ignored by the analysis parser, but can be targeted in % SurrogateTraining.

```
YoungsModulus $YM 2.0e8

% SurrogateTraining
@Parameter: $YM Material Uniform 1.0e8 3.0e8
%%%
```

1.1.5 Whitespace & Case Rules (What Is Case-Insensitive vs Case-Sensitive)

- **Case-insensitive** (you can use any capitalization):
 - Section headers like % Nodes, % Materials, % Step Definitions
 - Directive keys like @UMAT, @Perm, @PhaseChar, @Include, @NumRuns
- **Typically case-sensitive** (match exactly as written in the file):
 - Material IDs (e.g., Mat1)
 - Many parameter names *inside* a line (e.g., YoungsModulus, PoissonRatio, UMAT parameters like E, nu)
 - File paths (parsed as strings; case-sensitivity depends on your filesystem)

1.1.6 Whitespace Normalization (How Spaces Are Dealt With)

- Extra spaces/tabs are fine: tokens are separated by whitespace in most sections.
- The section delimiter %%% may have surrounding whitespace (e.g., %%%).
- For directives, : spacing is flexible:
 - @Perm:Constant k_sat 1e-7
 - @Perm: Constant k_sat 1e-7
 - @Perm : Constant k_sat 1e-7

1.1.7 Comments

Comments use the # character. Inline comments are supported:

```
% AnalysisType      # This selects the analysis type
PLNonCoupled
%%%
```

To keep a literal # character in a value, escape it as \#.

1.1.8 Section Markers

Sections are defined using the % marker followed by the section name:

```
% SectionName
[section content]
%%%
```

- **Section Start:** Each section begins with % followed by the section name (e.g., % Analysis Type, % Materials, % Boundary Conditions)
- **Section End:** Sections end with %%% (three percent signs) or when the next % section marker is encountered
- **Section Names:** Section names are **case-insensitive**, and whitespace/_/- differences are ignored. These are equivalent:
 - % Boundary Conditions
 - %BoundaryConditions
 - % boundary_conditions
 - % BOUNDARY-CONDITIONS

1.1.9 Include (Modular Files)

You can include external files using either form (case-insensitive):

```
% Include ./common_materials.txt
@Include: "./path with spaces/materials.txt"
```

Paths may be quoted with '...' or "..." to preserve spaces.

1.1.10 Path Resolution (Working / Project Directory)

The input parser reads file paths as plain strings; the solver then resolves them using the project/work directory rules described in [Command-Line Paths & Working Directory](#).

In general:

- Most **relative paths** referenced inside the input file are resolved relative to the resolved **project/work directory**.

- **Include** paths (`% Include ... / @Include ...`) are resolved relative to the directory of the file that contains the include first, then fall back to the project/work directory.

1.1.11 Property Markers

Many sections contain **directives** (sometimes called “properties”) that start with the @ marker:

```
@DirectiveName: [parameters]
```

- **Case-insensitive:** @Perm, @PERM, @perm are treated the same for directive lookup.
- **Colon spacing is flexible:** @Perm:Constant ... and @Perm: Constant ... both work.
- **Colon may be omitted** for directive headers that are followed by arguments (e.g., @Include path/to/file), but the recommended style is to include : when the directive reads like a key/value pair.
- **Recommended style:** @DirectiveName: <parameters> with a single space after :.
- **Leading @ count is ignored for lookup:** directive keys treat one-or-more leading @ the same, so @Key, @@Key, and @@@Key are equivalent (including for sub-keys inside blocks).
- **Recommended style for readability:** use @BlockHeader ... to start a block and @@SubKey: ... for lines inside the block; the parser accepts either @ or @@ for the same key.

Important (Materials section): Within % Materials, each property line must start with @ (e.g., @UMAT: ..., @Perm: ..., @PhaseChar: ...). This keeps material parsing unambiguous and consistent.

1.1.12 Multiple Definitions (Repeating Lines and Blocks)

Many sections allow you to define **multiple items** by repeating a line or repeating a block header:

- **Line-based sections:** each non-empty line is one item. Repeat the line format as many times as needed.
 - Examples: % Nodes, % Elements, % BoundaryConditions.
- **Block-based sections:** each item starts with a block header like @Name <id> ... and ends at the next block header (or the end of the section).
 - Examples: % Step Definitions (@Step <id>), % RigidBodies (@RigidBody <id>), % ContactPairs (@ContactPair <id>), % Ties (@Tie <id>), % Unties (@Untie <id>).
- **Single-block sections:** the section configures one item. To define more than one, repeat the entire % SectionName ... %% section later in the input file.
 - Examples: % BodyForce, % ReleaseBoundary, % RestrainDOFs.

General guidance: - Use **unique IDs** for block headers (@Step, @RigidBody, @Contact

Pair, ...). - When a section allows repeated lines/blocks, **order is usually not important**, except when an item references another item by ID (for example, elements reference node IDs and material IDs).

1.2 Core Required Sections

The input parser enforces the order of the **core mesh/step sections** (across the fully expanded input, including any % Include files):

1. **Analysis Type** (% AnalysisType) - Defines physics and coupling level
2. **Nodes** (% Nodes) - Mesh node coordinates
3. **Materials** (% Materials) - Material properties and IDs
4. **Elements** (% Elements) - Mesh connectivity and element types
5. **Step Definitions** (% Step Definitions) - Solver settings and time control

In addition, the following section is **required**:

6. **Boundary Conditions** (% Boundary Conditions) - Nodal DOF restraints (can be empty)

Important (order enforced): The ordering constraint applies to the core sequence: % AnalysisType → % Nodes → % Materials → % Elements → % Step Definitions. If any of these core sections appears out of order (e.g. % Elements before % Materials), FALCON reports a data format error indicating the expected sequence.

Important (boundary conditions required): % Boundary Conditions is required even if you have no restraints (use an empty block).

Important (infinite elements placement): If you use % Infinite Elements, it must appear **after** % Elements and **before** % Boundary Conditions.

1.2.1 Checkpoint / restart (optional)

- **Restart From** (% RestartFrom) - Optional; selects a checkpoint file/step to restart from
- **Checkpoint Control** (% CheckpointControl) - Optional; configures checkpoint saving

Recommended placement: put these immediately after % AnalysisType (or at least before % Step Definitions) so they are easy to find.

1.3 Optional Sections (can appear in many orders)

1.3.1 Constraints & modifiers

- **Release Boundary** (% Release Boundary) - Remove restraints mid-simulation
- **Restrain Dofs** (% Restrain Dofs) - Add restraints mid-simulation
- **MPC Constraints** (% MPCConstraints) - Multi-point constraints (e.g., slide along an arbitrary direction)

1.3.2 Loading & Forces

- **Stress Boundary** (% Stress Boundary) - Traction/pressure boundaries
- **Point Loads** (% PointLoads) - Concentrated nodal forces
- **Discharge Boundary Conditions** (% DischargeBoundaryConditions) - Flow boundaries
- **Body Force** (% Body Force) - Gravity only
- **Prescribed Values** (% Prescribed Values) - Nodal Dirichlet boundary conditions (e.g., U, p_w, p_g)

1.3.3 Wave Propagation & Mesh Control

- **Infinite Elements** (% Infinite Elements) - Mapped infinite elements for far-field truncation (must appear before % Boundary Conditions)
- **PML Layer** (% PML Layer) - Perfectly matched layer absorbing boundary (dynamic; must appear after % Elements)
- **Sponge Layer** (% SpongeLayer) - Absorbing boundaries for wave problems
- **ALE Configuration** (% ALE Configuration) - Arbitrary Lagrangian-Eulerian mesh smoothing for large deformation analyses

ARTEMIS DEV

1.3.4 Advanced Features

- **Contact Pairs** (% Contact Pairs) - Surface interaction definitions
- **Rigid Bodies** (% Rigid Bodies) - Rigid body constraints
- **Ties** (% Ties) - DOF coupling constraints
- **Unties** (% Unties) - Remove coupling constraints
- **Rigid Motion Constraints** (% RigidMotionConstraints) - Prescribed rigid motions

Recommended full input order (when you use everything) The parser only enforces the **core** ordering, but if you assume a “maximal” input file (core + most optional features), the following order is a clear, low-surprise layout that also respects placement constraints such as % Infinite Elements before % Boundary Conditions:

Suggested order	Section header	Required	Notes
1	% Analysis Type	Yes	Must be first core section.
2	% RestartFrom	No	If used, typically place near top.
3	% Checkpoint Control	No	If used, typically place near top.
4	% Nodes	Yes	Core section.
5	% Materials	Yes	Core section.

Suggested order	Section header	Required	Notes
6	% Elements	Yes	Core section; defines connectivity and principal nodes.
7	% MPC Constraints	No	Multi-point constraints (must appear after % Elements).
8	% Infinite Elements	No	If used, must appear after % Elements and before % Boundary Conditions.
9	% PML Layer	No	Must appear after % Elements (dynamic absorbing boundary).
10	% SpongeLayer	No	Must appear after % Elements (dynamic absorbing boundary).
11	% ALE Configuration	No	Must appear after % Elements (mesh smoothing).
12	% Step Definitions	Yes	Core section (must be after % Elements).
13	% Initial Assignments	No	If used, often placed after steps for clarity.

Suggested order	Section header	Required	Notes
14	% Step Initial Assignments	No	Step-triggered initial-field overrides (applied at step start; overwrites current Gauss-point state).
15	% Step Materials	No	Step-start element material reassignment (overwrites % Elements material IDs for selected elements, referencing materials defined in % Materials).
16	% Boundary Conditions	Yes	Required (can be empty). Recommended after % Elements and after % Infinite Elements if used.
17	% Release Boundary / % Restrain Dofs	No	Constraint changes during the run (step-controlled).
18	% Contact Pairs	No	Define contact interfaces (if used).
19	% Rigid Bodies	No	Define rigid bodies (if used).
20	% Ties / % Unties	No	Tie/untie DOFs (if used).

Suggested order	Section header	Required	Notes
21	% RigidMotion Constraints	No	Prescribed rigid motions (if used).
22	% Body Force	No	Gravity, etc.
23	% Stress Boundary	No	Tractions/pressures.
24	% PointLoads	No	Concentrated nodal forces.
25	% Discharge Boundary Conditions	No	Flow discharge BCs (coupled/-fully coupled).
26	% Prescribed Values	No	Time-varying prescribed values.
27	Output/monitoring sections	No	% DOFOutput, % ReactionForce Sum, % Point StateOutput, % LineState Output, etc.
28	% Element Activity	No	Element activation/deactivation schedules.
29	% PostStep Actions	No	Post-step actions.
30	% Surrogate Training	No	Parametric studies / surrogate training.

Notes: - % Include / @Include can appear anywhere, but core-order validation applies to the **expanded** file after includes. - Many optional sections can legally appear in other places; the table above is a recommended organization for readability.

1.3.5 Initialization

- **Initial Assignments** (% Initial Assignments) - Set initial field variables
- **Step Initial Assignments** (% Step Initial Assignments) - Step-triggered initial-field overrides (applied at step start; overwrites the current Gauss-point state)
- **Step Materials** (% Step Materials) - Overwrite the material assignment of selected elements at step start (reference a material defined in % Materials)

1.3.6 Output & Monitoring

- **Reaction Force Sum** (% ReactionForceSum) - Output reaction forces
- **DOF Output** (% DOFOutput) - Custom DOF monitoring
- **Point State Output** (% PointStateOutput) - Interpolate state variables at arbitrary coordinates (see dedicated chapter)
- **Line State Output** (% LineStateOutput) - Sample state variables along user-defined straight lines (see dedicated chapter)
- **Picard Loop Output** (% PicardLoopOutput) - Output Picard-loop statistics during unsaturated initialization
- **Master Force Contact** (% MasterForceContact) - Output contact forces for selected contact pairs
- **Element Activity** (% ElementActivity) - Activate/deactivate elements
- **Post Step Actions** (% PostStepActions) - Actions after simulation steps

1.3.7 Parametric Studies & Uncertainty Quantification

- **Surrogate Training** (% SurrogateTraining) - Parametric analysis, uncertainty quantification, and surrogate model training through automated ensemble simulations

1.3.8 Modular Files

- **Include** (% [Include](#) / [@Include](#)) - Include external files

1.4 Summary of Input Sections

The table below summarizes all top-level section headers that the input system recognizes. It also notes which analysis types or element/DOF sets each section is primarily intended for.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Analysis Type	Select overall physics and coupling (PL/AX/3D, coupled/uncoupled)	All	Global	Required; core section, ordering enforced.
% Restart From	Load simulation state from checkpoint file	All	Global	Optional; may appear anywhere; typically placed near the top after % AnalysisType.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Checkpoint Control	Configure checkpoint saving behavior	All	Global	Optional ; may appear anywhere; typically placed near the top.
% Nodes	Define node IDs and coordinates	All	All elements	Required ; coordinates interpreted as 2D or 3D based on analysis type.
% Materials	Define material IDs and constitutive properties	All	All elements	Required ; materials must cover all element material IDs.
% Elements	Define element connectivity and types	All	Element types N3/N4/N6/N8 with variants C/F	Required ; element type must match DOFs implied by analysis (coupled / fully).
% Boundary Conditions	Apply nodal DOF restraints (Dirichlet BCs)	All	Any displacement / pressure DOFs allowed by analysis type	Required (can be empty); needed to initialize active DOFs and global indexing.
% Stress Boundary	Apply traction-pressures on element edges/-faces	All mechanical / coupled analyses	Elements with appropriate boundary faces (2D: tri/quad edges; 3D: quad/hex faces)	Loads are integrated over edges (2D) or faces (3D).
% Point Loads	Apply concentrated nodal forces	All	Displacement DOFs (DisX, DisY, DisZ in 3D)	Applied directly to nodes; ignored on re-restrained/inactive DOFs.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Discharge Boundary Conditions	Apply flow (discharge) boundary conditions	Coupled / fully coupled analyses	Flow DOFs (PW, PA)	Also accepts % Discharge Boundary / % Discharge Boundaries spellings.
% Step Definitions	Configure time steps, solvers, outputs	All	Global	Required; defines simulation steps and output frequencies.
% Initial Assignments	Set initial state variables / pore pressures / void ratios	All	Gauss points / nodes depending on assignment mode	Often used for geostatic or hydrostatic initialization.
% Step Initial Assignments	Step-triggered initial-field overrides	All	Gauss points (step-start overwrite)	Applied at step start (before birth/kill and before the Newton solve).
% Step Materials	Overwrite element material IDs at step start (reference an existing material)	All	Elements listed in the block	Optional; @@MaterialId must exist in % Materials; applied at step start.
% Sponge Layer	Define absorbing “sponge” boundaries for wave problems	Dynamic analyses	Boundary nodes / elements	Primarily for soil dynamics / wave propagation to reduce reflections.
% PML Layer	Define a perfectly matched layer (PML) absorbing boundary	Dynamic analyses	Boundary elements/nodes on selected sides	Accepts % PMLayer / % PML-Layer spellings; must appear after % Elements.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Infinite Elements	Define mapped infinite elements for far-field truncation	All	Boundary regions where infinite elements are attached	Must appear before % Boundary Conditions (DOF setup depends on it).
% MPC Constraints	Define multi-point constraints (e.g., slide along an arbitrary direction)	All	Displacement DOFs	Must appear after % Elements; can be activated/deactivated using % RestrainDOFs / % Release Boundary with Constraint: lines.
% Contact Pairs	Define master/slave contact surfaces	2D PL/AX analyses with contact (no 3D contact support)	Elements forming contact interfaces	Requires matching 2D element types (same interpolation order) on master and slave surfaces; contact-specific Gauss points are used along shared edges.
% Release Boundary	Release (unrestrain) previously restrained DOFs at a step	All	Node DOFs	Used for staged construction / removal of supports.
% Restrain Dofs	Add new DOF restraints mid-analysis	All	Node DOFs	Opposite of % Release Boundary; for activating restraints later in time.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Rigid Bodies	Define rigid bodies and their properties	Analyses with rigid inclusions	Nodes/elements assigned to rigid bodies	Used for contact problems involving structural elements, such as steel or concrete, that can be reasonably assumed rigid relative to the surrounding soil.
% Ties	Tie slave nodes DOFs to a reference node	All	Displacement / pressure DOFs depending on analysis	Used for multi-block connectivity, symmetry, or interface tying.
% Unties	Remove previously defined tie constraints	All	Tied DOFs	Allows staged deactivation of ties.
% Prescribed Values	Time-varying prescribed displacements/velocities/accelerations	All	Displacement DOFs (and pressure rate in coupled cases)	Used for kinematic loading, moving boundaries, etc.
% Body Force	Define body forces (e.g., gravity, seismic base motion)	All	All active elements	For coupled analyses, contributions may be split by phase (solid/liquid/gas).
% Rigid Motion Constraints	Define rigid motion constraints via algebraic expressions	All	Selected nodes / DOFs via referenced rigid bodies	Supports parametric and tabular rigid motions (translation / rotation) applied to rigid bodies.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Reaction ForceSum	Output summed reaction forces at selected nodes/DOFs	All	Any restrained DOFs (DisX/Y/Z, PW, PA)	Postprocessing only; does not affect solution.
% DOFOutput	Output time histories of DOFs at specified node IDs	All	Displacement & pressure DOFs (DisX/Y/Z, PW, PA)	Postprocessing only; CSV output per DOF and node.
% Point StateOutput	Interpolate state variables at arbitrary coordinates	All	State variables defined at nodes (after stress recovery)	Supports stress, strain, pore pressures, etc.; see <i>Point & Line State Output</i> chapter.
% LineState Output	Interpolate state variables along straight lines	All	Same as % Point StateOutput , sampled along user-defined lines	Supports step-based and time-based triggering with different CSV layouts.
% Picard LoopOutput	Output Picard-loop statistics during unsaturated initialization	Fully coupled unsaturated analyses	Gauss points where Picard loop runs	Only active for certain models that require post-equilibrium state conditioning, such as GCC.
% ALE Configuration	Configure ALE mesh smoothing for large deformation analyses	Large deformation analyses	Interior nodes / elements	Prevents mesh distortion in large deformation problems.
% Master Force Contact	Output contact forces for selected contact pairs	Analyses with contact	Contact pairs / nodes	Postprocessing only; useful for load transfer diagnostics.

Section header	Purpose	Analysis types	Element / DOF applicability	Notes / considerations
% Element Activity	Activate / deactivate elements at specific steps	All	Elements listed in birth/kill schedule	For staged construction, excavation, and deactivation.
% PostStep Actions	Execute special actions after certain steps	All	Elements / materials / constraints selected in actions	Used to change material models, properties, or other settings mid-analysis.
% Surrogate Training	Parametric analysis, UQ, and surrogate model training	All	Global	Runs multiple simulations with varied parameters; generates CSV dataset for ML training or UQ analysis.
% Include / @ Include	Include external input files	All	Global	Paths can be relative or absolute; cycles are detected and rejected.

1.5 Analysis Type

Defines the overall physics and coupling level.

```
% AnalysisType
PLNonCoupled
%%%
```

Supported Analysis Types:

Plane Strain (PL): - PLNonCoupled or PLUnCoupled - Single-phase mechanical analysis - PLCoupled - Bi-phase (solid + water) analysis - PLFullyCoupled - Multi-phase (solid + water + air) analysis

Axisymmetric (AX): - AXNonCoupled or AXUnCoupled - Single-phase mechanical analysis -

AXCoupled - Bi-phase (solid + water) analysis - AXFullyCoupled - Multi-phase (solid + water + air) analysis

3D: - ThreeDNonCoupled or ThreeDUnCoupled - Single-phase mechanical analysis - ThreeDCoupled - Bi-phase (solid + water) analysis - ThreeDFullyCoupled - Multi-phase (solid + water + air) analysis

FALCON supports 2D (plane strain, axisymmetric) and 3D analyses. The analysis type determines the dimensionality and available element types.

1.6 Nodes

Lists all mesh nodes with coordinates.

```
% Nodes
1 0.0 0.0
2 1.0 0.0
3 1.0 1.0
4 0.0 1.0
%%%
```

Format: - <NodeID> <X> <Y>

1.7 Materials

Defines material IDs and their constitutive properties. FALCON uses **UMAT (User Material)** as the primary method for defining constitutive models, allowing flexible implementation of mechanical, SWRC, effective stress, and permeability models through external C++ files.

```
% Materials
Mat1
@UMAT:/path/to/LinearElasticUMAT.cpp /path/to/LinearElasticUMAT.hpp
Mechanical YoungsModulus=1000000 PoissonsRatio=0.01
@PhaseChar: Solid rhos 2.7
%%%
```

Format: - Start with a material ID (string identifier) - @<PropertyType>: lines follow, possibly multiple per material

See the [Materials](#) chapter for complete details on: - UMAT syntax and categories - Built-in property syntax (@Perm:, @SWRC:, @EffectiveStress:, @PhaseChar:, @AnisotropicPerm:

) - Material requirements by analysis type - Supported models and parameters - Error codes and troubleshooting

1.8 Elements

Specifies mesh connectivity and element types.

```
% Elements
1 N4P4C 1 2 3 4 Mat1
2 N4P4C 5 6 7 8 Mat2
%%%
```

Format:

- <ElementID> <ElementType> <NodeID1> ... <NodeIDN> <MaterialID>
- Element types follow the pattern N<X>P<Y>[C|F] where:
 - N<X> = number of nodes (3, 4, 6, 8)
 - P<Y> = number of integration points (1, 3, 4, 6, 8, 9)
 - C = coupled analysis (includes pore pressure DOFs)
 - F = fully coupled analysis (includes pore water + air pressure DOFs)

- **Supported Element Types:**

- N6P6, N3P3, N4P4, N8P8 (uncoupled elements)
- N6P6C, N8P8C (coupled elements)
- N6P6F, N8P8F (fully coupled elements)

- Nodes ordered counterclockwise for proper element orientation.
-

1.9 Initial Assignments

Set initial field variables before loading using height-based assignments or external coordinate files.

```
% Initial Assignments
@Stress: H 0.0 values -36.0 -60.0 -36.0 0.0 0.0 0.0 H 10.0 values
-50.0 -80.0 -50.0 0.0 0.0 0.0
@PW: H 0.0 values -10.0 H 10.0 values -30.0
```

```
@Void:      H 0.0   values 0.45   H 10.0   values 0.50
@CoordinateFile /path/to/coords.txt
%%%
```

See the [Initial Assignments](#) chapter for complete details on all available options.

1.10 Step Definitions

Define solver settings and time control for simulation steps.

```
% Step Definitions
@Step 1
@@StartStep: 0
@@StepTime: 100.0
@@NumberSteps: 10
@@SolverType: Direct
@@OutputControlType: ByStep
@@OutputControlValue: 1
@@OutputTypes: Displacement EffStress VoidRatio
@@AutomaticStepControl: No
@@ErrorTarget: 1e-3
%%%
```

See the [Steps](#) chapter for complete details on all step configuration options.

1.11 Additional Sections Reference

The following sections can appear in any order after the core sections. Each section starts with a % marker and ends with %% or the next % section marker.

1.11.1 Boundary Conditions & Constraints

- [% Boundary Conditions](#) - DOF restraints and boundary conditions
- [% MPCConstraints](#) - Multi-point constraints (e.g., sliding along a direction)
- [% Release Boundary](#) - Remove DOF restraints mid-simulation
- [% Restrain Dofs](#) - Add DOF restraints mid-simulation

1.11.2 Loading & Forces

- [% Stress Boundary](#) - Traction/pressure boundary conditions
- [% PointLoads](#) - Concentrated nodal forces

- % [DischargeBoundaryConditions](#) - Flow boundary conditions
- % [Body Force](#) - Body forces (gravity, seismic loading)
- % [Prescribed Values](#) - Time-varying prescribed values

1.11.3 Wave Propagation & Mesh Control

- % [SpongeLayer](#) - Absorbing boundaries for wave problems
- % [PML Layer](#) - Perfectly matched layer (PML) absorbing boundary
- % [Infinite Elements](#) - Mapped infinite elements for far-field truncation
- % [ALE Configuration](#) - Arbitrary Lagrangian-Eulerian mesh smoothing for large deformation analyses

1.11.4 Advanced Features

- % [Contact Pairs](#) - Surface interaction definitions
- % [Rigid Bodies](#) - Rigid body constraints
- % [Ties](#) - DOF coupling constraints
- % [Unties](#) - Remove coupling constraints
- % [RigidMotionConstraints](#) - Prescribed rigid motions

1.11.5 Output & Monitoring

- % [ReactionForceSum](#) - Output reaction forces
- % [DOFOutput](#) - Custom DOF monitoring
- % [PointStateOutput](#) - Interpolate state variables at arbitrary coordinates
- % [LineStateOutput](#) - Sample state variables along user-defined straight lines
- % [PicardLoopOutput](#) - Output Picard-loop statistics during unsaturated initialization
- % [MasterForceContact](#) - Contact force output
- % [ElementActivity](#) - Activate/deactivate elements
- % [PostStepActions](#) - Actions after simulation steps

1.11.6 Parametric Studies & Uncertainty Quantification

- % [SurrogateTraining](#) - Parametric analysis, uncertainty quantification, and surrogate model training through automated ensemble simulations

1.11.7 Modular Files

- **Include** (% [Include](#) / [@Include](#)) - Include external files for modular input organization
 - Supports relative and absolute paths
 - Prevents cyclic includes automatically
 - Useful for organizing large input files into manageable sections

1.12 Error Codes and Troubleshooting

FALCON's input reader provides detailed error messages with unique error codes to help diagnose input file issues. For a complete reference of all error codes, their meanings, and resolution strategies, see the dedicated [Error Dictionary](#).

