AD FALCON API Manual

Prescribed Values

Javad Ghorbani

March 26, 2026

Contents

1 Prescribed Values

The **Prescribed Values** section defines specific conditions for nodes in the simulation, such as displacement, velocity, or acceleration, applied over time using various load types. These values can be constant, time-dependent, or defined using a table of values.

1.1 Syntax

1.2 Section header

FALCON treats section names as case-insensitive and whitespace-insensitive, so these headers are equivalent:

```
% PrescribedValues
% Prescribed Values
% prescribed_values
```

1.3 Section format

```
% PrescribedValues
@PrescribedValue <Type> <ID>
  @@DOF: <DOFName>
  @@Amplitude: <Value>
  @@LoadType: <Immediate|Ramp|Sinusoidal|DampedSinusoidal|Tabular>
  @@StartStep: <StepID>
  @@Frequency: <Frequency>             # required for sinusoidal types
  [@@DampingFactor: <DampingFactor>]   # damped sinusoidal
  [@@PhaseLag: <Degrees>]              # sinusoidal types
  @@NodeIDs: <n1> <n2> ...
  [@@TabularData: <t1> <m1>; <t2> <m2>; ...]   # for LoadType Tabular
  @@Propagate: Yes | FinalStep <N>     # For LoadType Tabular, only Yes is
supported

@PrescribedValue <Type> <ID>
  ...

%%%   # end of % PrescribedValues (or start the next % section)
```

Notes:

- `%%%` ends the `% PrescribedValues` section. For multiple prescribed values, do not put `%%%` between blocks; start a new `@PrescribedValue ...` header instead.

- `LoadType` names are case-insensitive (e.g. `ramp`, `Ramp`).
- `Type` in `@PrescribedValue <Type> <ID>` is case-insensitive (e.g. `displacement`, `Velocity`).
- `@PrescribedValue` IDs accept an optional trailing `:` (so `@PrescribedValue Displacement 1:` works).
- Directive names are case-insensitive and tolerate one or more leading `@` characters (recommended style: `@@Key: ...`).
- `@@TabularData:` uses semicolons (`;`) to separate points; each point is `time magnitude`.

1.4 Multiple prescribed values

- Define multiple prescribed values by repeating `@PrescribedValue <Type> <ID>` blocks inside the same `% PrescribedValues` section.
- IDs should be unique within the section; the next `@PrescribedValue ...` line begins a new block.

For available `LoadType` options, definitions, and equations, see Load Types.

1.5 Key Components

1. **Type of Prescribed Value**:
   - Specifies the physical property being controlled:
   - `Displacement`: Prescribes node displacements.
   - `Velocity`: Prescribes node velocities.
   - `Acceleration`: Prescribes node accelerations.
   - `PressureRate` (alias `PressureDot`): Prescribes a pressure rate on the selected DOF.
   - Example:

```
@PrescribedValue Displacement 1
```

2. **DOF (Degree of Freedom)**:
   - Indicates which degree of freedom is controlled (e.g., `DisX`, `DisY`, `DisZ`).
3. **Amplitude**:
   - The magnitude of the prescribed value.
4. **LoadType**:
   - Specifies the type of variation for the prescribed value over time (see Load Types):
   - `Immediate`: Applied instantly.
   - `Ramp`: Linearly increases from 0 to the specified value.

- `Sinusoidal`: Varies sinusoidally with time.
- `DampedSinusoidal`: Sinusoidal variation with decreasing amplitude.
- `Tabular`: Defined using tabular data.
- Example:

```
@@LoadType: Ramp
```

5. **StartStep**:
   - Specifies the step where the prescribed value begins.
6. **Frequency, DampingFactor, and PhaseLag**:
   - Additional parameters for sinusoidal or damped sinusoidal load types.
   - `PhaseLag` is specified in degrees.
7. **Node IDs**:
   - The list of nodes where the prescribed value is applied.
   - Example:

```
@@NodeIds: 1 2 3
```

8. **TabularData**:
   - Time-magnitude pairs for tabular load types.
   - Example:

```
@@TabularData: 0 0; 1 10; 2 20
```

9. **Propagate**:
   - Determines if the condition applies to subsequent steps:
   - `Yes`: Applies to all steps after `StartStep`.
   - `FinalStep <StepNumber>`: Applies up to the specified step.

1.6 Example Input

1.7 Displacement with Ramp Load

```
% Prescribed Values
@PrescribedValue Displacement 1
@@DOF: DisX
@@Amplitude: 10
@@LoadType: Ramp
@@StartStep: 1
@@NodeIds: 1 2 3
@@Propagate: Yes
%%%
```

1.8 Sinusoidal Velocity

```
% Prescribed Values
@PrescribedValue Velocity 2
@@DOF: DisY
@@Amplitude: 5
@@LoadType: Sinusoidal
@@Frequency: 1.0
@@PhaseLag: 0.5
@@StartStep: 2
@@NodeIds: 4 5 6
@@Propagate: FinalStep 4
%%%
```

1.9 Tabular Acceleration

```
% Prescribed Values
@PrescribedValue Acceleration 3
@@DOF: DisZ
@@Amplitude: 1.0
@@LoadType: Tabular
@@TabularData: 0 0; 1 10; 2 20; 3 30
@@StartStep: 1
@@NodeIds: 7 8 9
@@Propagate: Yes
%%%
```

1.10 Validation Rules

1. **Type Validation**:

- Supported types: `Displacement`, `Velocity`, `Acceleration`, `PressureRate` (alias `PressureDot`).

2. **DOF Validation**:

- The `DOF` must correspond to an allowable degree of freedom for the node.

3. **Load Type Validation**:

- Parameters must match the specified `LoadType`:
  - `Immediate` requires only `Amplitude`.
  - `Ramp` requires `Amplitude`.
  - `Sinusoidal` requires `Amplitude` and `Frequency` (with `Frequency > 0`); `PhaseLag` is optional.
  - `DampedSinusoidal` requires `Amplitude`, `Frequency` (with `Frequency > 0`), and `DampingFactor` (with `DampingFactor >= 0`); `PhaseLag` is optional.
  - `Tabular` requires `TabularData`.

4. **Node Validation**:

- All `NodeIds` must exist in the mesh.

5. **StartStep Validation**:

- `StartStep` must reference a defined simulation step ID (and must be >= 1).

6. **Propagation Validation**:

- `Propagate` is required. For `LoadType: Tabular`, `Propagate` must be `Yes` (not `Final Step`).

1.11 Summary

The **Prescribed Values** section allows precise control over node behavior during the simulation. With options for displacement, velocity, and acceleration, and flexibility in load application methods, it supports diverse scenarios in static and dynamic analyses. Proper validation ensures the integrity of input data, minimizing errors during execution.