AD FALCON API Manual

Tie/Untie

Javad Ghorbani

March 26, 2026

Contents

1 Tie/Untie

The **Tie** / **Untie** sections let you temporarily merge (tie) selected DOFs of multiple nodes, and later restore (untie) those DOFs.

1.1 Syntax

1.1.1 Section headers

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

```
% Ties
% ties
% tie_s
```

```
% Unties
% unties
% un_ties
```

1.1.2 Tie blocks

Inside `% Ties`, define one or more `@Tie` blocks. The `% Ties` section ends at `%%%` (or when the next `% ...` section begins).

```
% Ties
@Tie <ID>
  @@ReferenceNode: <nodeId>
  @@ReferenceDOF: <DOF1> <DOF2> ...      # or @@ReferenceDOFs:
  @@TiedNodes: <n1> <n2> ...
  @@Steps: <s1> <s2> <a-b> <c:d> ...

@Tie <ID>
  ...
%%%
```

1.1.3 Untie blocks

Inside `% Unties`, define one or more `@Untie` blocks. The `% Unties` section ends at `%%%` (or when the next `% ...` section begins).

```
% Unties
@Untie <ID>
  @@TieID: <tieId>
  @@Steps: <s1> <s2> <a-b> <c:d> ...

@Untie <ID>
  ...
%%%
```

Notes:

- Directive names are case-insensitive and tolerate one or more leading `@` characters.
- Node/step lists accept spaces, commas, and semicolons; ranges may be written as `a-b` or `a:b`.
- `@Tie <ID>` / `@Untie <ID>` also accept an optional trailing `:` (e.g. `@Tie 1:`).

1.2 Tie

A **Tie** merges the listed DOFs of each slave node into the same unknown as the reference node DOF.

1.2.1 Keys

- `@@ReferenceNode:` master node ID.
- `@@ReferenceDOF:` or `@@ReferenceDOFs:` one or more DOF names.
- `@@TiedNodes:` slave node IDs.
- `@@Steps:` step IDs where the tie is active.

1.2.2 Validation

- Tie `ID` must be unique.
- `ReferenceNode` and `TiedNodes` must exist.
- DOFs must be valid for the analysis and must be **active (not restrained)** on the reference and all slave nodes.
- All `Steps` must refer to defined `@Step` IDs.

1.3 Untie

An **Untie** restores the tied slave DOFs back into independent unknowns at the specified steps.

1.3.1 Keys

- `@@TieID:` ID of the tie to release.
- `@@Steps:` step IDs where the untie is applied.

1.3.2 Validation

- `TieID` must refer to an existing `@Tie`.
- All `Steps` must refer to defined `@Step` IDs.
- Untie steps must occur strictly after the tie's earliest active step.

1.4 Typical usage

Because tying collapses the DOF structure, an untie is usually scheduled on the **first step after the last step where the tie is active**.

1.5 Example

```
% Ties
@Tie 10
  @@ReferenceNode: 5
  @@ReferenceDOFs: DisX DisY
  @@TiedNodes: 8 12 15
  @@Steps: 2-4
%%%

% Unties
@Untie 20
  @@TieID: 10
  @@Steps: 5
%%%
```

- Tie 10 is active on steps 2–4.
- Untie 20 restores the tied DOFs at step 5.