

# Vector Coding

The **Vector Coding** command can be used to determine inter-segment coordination with the output being an angle from 0-360 degrees known as the coupling angle. All of the math behind **Vector Coding** is done internally with Visual3D, but a brief summary is provided here for user knowledge. This command is based on work from Needham 2015 with more detailed explanations available in the [References](#)

## Vector Coding Process

The Following section will outline the equations and steps followed while conducting the vector coding.

### Coupling Angle Calculation

The command is based on changes in consecutive segment angles for proximal and distal segments of a given joint. **Equation 1** and **Equation 2** are used to calculate the coupling angle, with the equation used determined by the difference in consecutive angles for the proximal segment. If this result is greater than zero **Equation 1** is used and if the result is less than zero **Equation 2** is used.

$$\gamma_i = \left( \frac{\theta_{D(i+1)} - \theta_{Di}}{\theta_{P(i+1)} - \theta_{Pi}} \right) * \frac{180}{\pi} \quad \theta_{P(i+1)} - \theta_{Pi} > 0 \quad (1)$$

$$\gamma_i = \left( \frac{\theta_{D(i+1)} - \theta_{Di}}{\theta_{P(i+1)} - \theta_{Pi}} \right) * \frac{180}{\pi} + 180 \quad \theta_{P(i+1)} - \theta_{Pi} < 0 \quad (2)$$

### Angle Conditions

**Equation 3** displays conditions that are applied to the angle to avoid inaccuracies from special cases where there is no change in angle at consecutive points.

$$\gamma_i \begin{cases} \gamma_i = 90 & \theta_{D(i+1)} - \theta_{Di} = 0 \text{ and } \theta_{P(i+1)} - \theta_{Pi} > 0 \\ \gamma_i = -90 & \theta_{D(i+1)} - \theta_{Di} = 0 \text{ and } \theta_{P(i+1)} - \theta_{Pi} < 0 \\ \gamma_i = -180 & \theta_{D(i+1)} - \theta_{Di} < 0 \text{ and } \theta_{P(i+1)} - \theta_{Pi} = 0 \\ \gamma_i = Undefined & \theta_{D(i+1)} - \theta_{Di} = 0 \text{ and } \theta_{P(i+1)} - \theta_{Pi} > 0 \end{cases} \quad (3)$$

## Angle Correction

The results from above must be corrected to a value between 0 and 360 degrees. **Equation 4** is used to make this correction.

$$\gamma_i \begin{cases} \gamma_i + 360 & \gamma_i < 0 \\ \gamma_i & \gamma_i \geq 0 \end{cases} \quad (4)$$

## Pipeline Command

The **Vector\_Coding** command can be found in the Pipeline Workshop within the **Signal Process** folder as so:

```
Vector_Coding
! /SIGNAL_OWNER1=
/SIGNAL_TYPE1=
/SIGNAL_FOLDER1=
/SIGNAL_NAME1=
! /SIGNAL_COMPONENT1=
! /SIGNAL_OWNER2=
/SIGNAL_TYPE2=
/SIGNAL_FOLDER2=
/SIGNAL_NAME2=
! /SIGNAL_COMPONENT2=
! /RESULT_OWNER=
! /RESULT_FOLDER= PROCESSED
/RESULT_NAME=
;
```

## Command Parameters

The parameters that can be used to control the command are as follows:

|                         |                                                            |
|-------------------------|------------------------------------------------------------|
| <b>Signal Owner</b>     | The name of the file containing the given signal           |
| <b>Signal Type</b>      | The type of signal (force, target, link_model_based, etc.) |
| <b>Signal Folder</b>    | The name of the folder containing the given signal         |
| <b>Signal Component</b> | Which component of the signal is used                      |
| <b>Result Owner</b>     | The name of the file that will contain the result          |
| <b>Result Folder</b>    | The name of the folder containing the resulting signal(s)  |
| <b>Result Name</b>      | The name of the resulting signal                           |

Understand the parameters of the Vector\_Coding command:

## SIGNAL1

The /SIGNAL\_OWNER1, /SIGNAL\_TYPE1, /SIGNAL\_FOLDER1, and /SIGNAL\_NAME1 parameters allow the user to specify which signal (by [file](#), [data type](#), [folder](#), and name) should be used as the first signal for the Vector\_Coding calculation. The proximal segment angle should be chosen as signal 1.

## SIGNAL2

Similarly, the /SIGNAL\_OWNER2, /SIGNAL\_TYPE2, /SIGNAL\_FOLDER2, and /SIGNAL\_NAME2 parameters allow the user to specify the second signal for the Vector\_Coding calculation. The distal segment angle should be chosen as signal 2.

## RESULT

The /RESULT\_OWNER, /RESULT\_FOLDER, and /RESULT\_NAME parameters allow the user to specify where the result of the vector coding should be saved.

- The result's data type the DERIVED folder.
- The result's folder defaults to being the PROCESSED folder.

## Example: Performing Vector Coding on the Ankle

```
Compute_Model_Based_Data
/RESULT_NAME= RIGHT_FOOT_ANGLE
/FUNCTION=SEG_PROGRESSION_ANGLE
/SEGMENT=Right Foot
! /REFERENCE_SEGMENT=LAB
! /RESOLUTION_COORDINATE_SYSTEM=LAB
! /USE_CARDAN_SEQUENCE=FALSE
! /NORMALIZATION=FALSE
! /NORMALIZATION_METHOD=
! /NORMALIZATION_METRIC=
! /NEGATEX=FALSE
! /NEGATEY=FALSE
! /NEGATEZ=FALSE
```

```
! /AXIS1=X
! /AXIS2=Y
! /AXIS3=Z
! /INCLUDE_REMOTE_ANGULAR_MOMENTUM=FALSE
! /TREADMILL_DATA=FALSE
! /TREADMILL_DIRECTION=UNIT_VECTOR(0,1,0)
! /TREADMILL_SPEED=0.0
;
```

#### Compute\_Model\_Based\_Data

```
/RESULT_NAME= RIGHT_SHANK_ANGLE
/FUNCTION=SEG_PROGRESSION_ANGLE
/SEGMENT= Right Shank
! /REFERENCE_SEGMENT=LAB
! /RESOLUTION_COORDINATE_SYSTEM=LAB
! /USE_CARDAN_SEQUENCE=FALSE
! /NORMALIZATION=FALSE
! /NORMALIZATION_METHOD=
! /NORMALIZATION_METRIC=
! /NEGATEX=FALSE
! /NEGATEY=FALSE
! /NEGATEZ=FALSE
! /AXIS1=X
! /AXIS2=Y
! /AXIS3=Z
! /INCLUDE_REMOTE_ANGULAR_MOMENTUM=FALSE
! /TREADMILL_DATA=FALSE
! /TREADMILL_DIRECTION=UNIT_VECTOR(0,1,0)
! /TREADMILL_SPEED=0.0
;
```

#### Vector\_Coding

```
/SIGNAL_OWNER1=*.c3d
/SIGNAL_TYPE1=LINK_MODEL_BASED
/SIGNAL_FOLDER1=ORIGINAL
/SIGNAL_NAME1=RIGHT_SHANK_ANGLE
/SIGNAL_COMPONENT1=X
/SIGNAL_OWNER2=*.c3d
/SIGNAL_TYPE2=LINK_MODEL_BASED
/SIGNAL_FOLDER2=ORIGINAL
/SIGNAL_NAME2=RIGHT_FOOT_ANGLE
/SIGNAL_COMPONENT2=X
/RESULT_OWNER=*.c3d
!/RESULT_FOLDER=PROCESSED
/RESULT_NAME=Ankle_Couple
;
```

## References

The **Vector Coding** process is based on articles by Robert Needham. Needham, R., Naemi, R., & Chockalingam, N. (2014). Quantifying lumbar-pelvis coordination during gait using a modified vector coding technique. Journal of Biomechanics, 47(5), 1020–1026.

<https://doi.org/10.1016/j.jbiomech.2013.12.032> [1]

From:

<https://has-motion.com/wiki/> - Wiki Documentation Site

Permanent link:

[https://has-motion.com/wiki/doku.php?id=visual3d:documentation:pipeline:signal\\_commands:vector\\_coding&rev=1779976077](https://has-motion.com/wiki/doku.php?id=visual3d:documentation:pipeline:signal_commands:vector_coding&rev=1779976077)

Last update: 2026/05/28 13:47

