ETH zürich

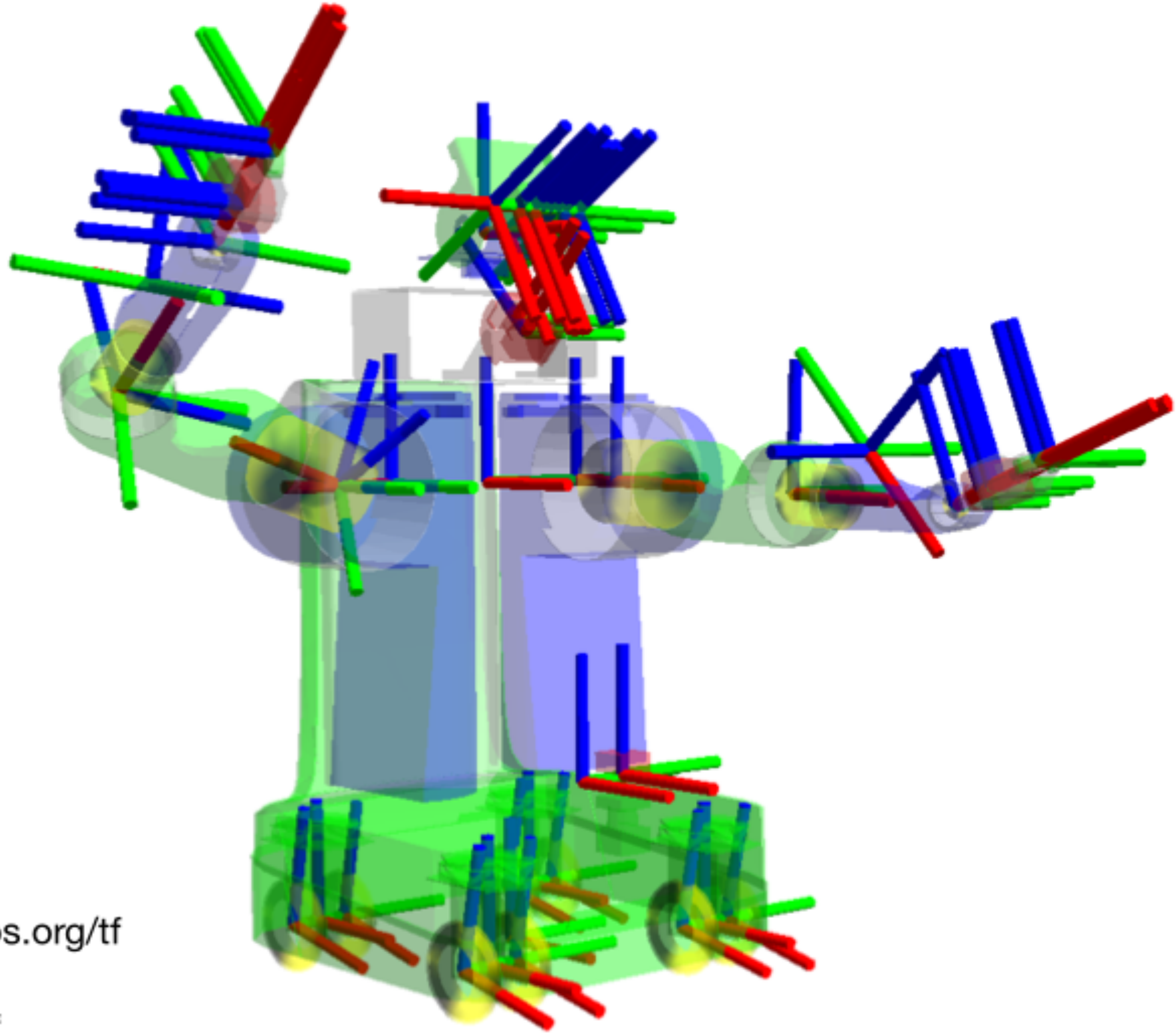
Representing Robot Pose

The Good, the Bad, and the Ugly

Paul Furgale

ETH Zürich

The Good



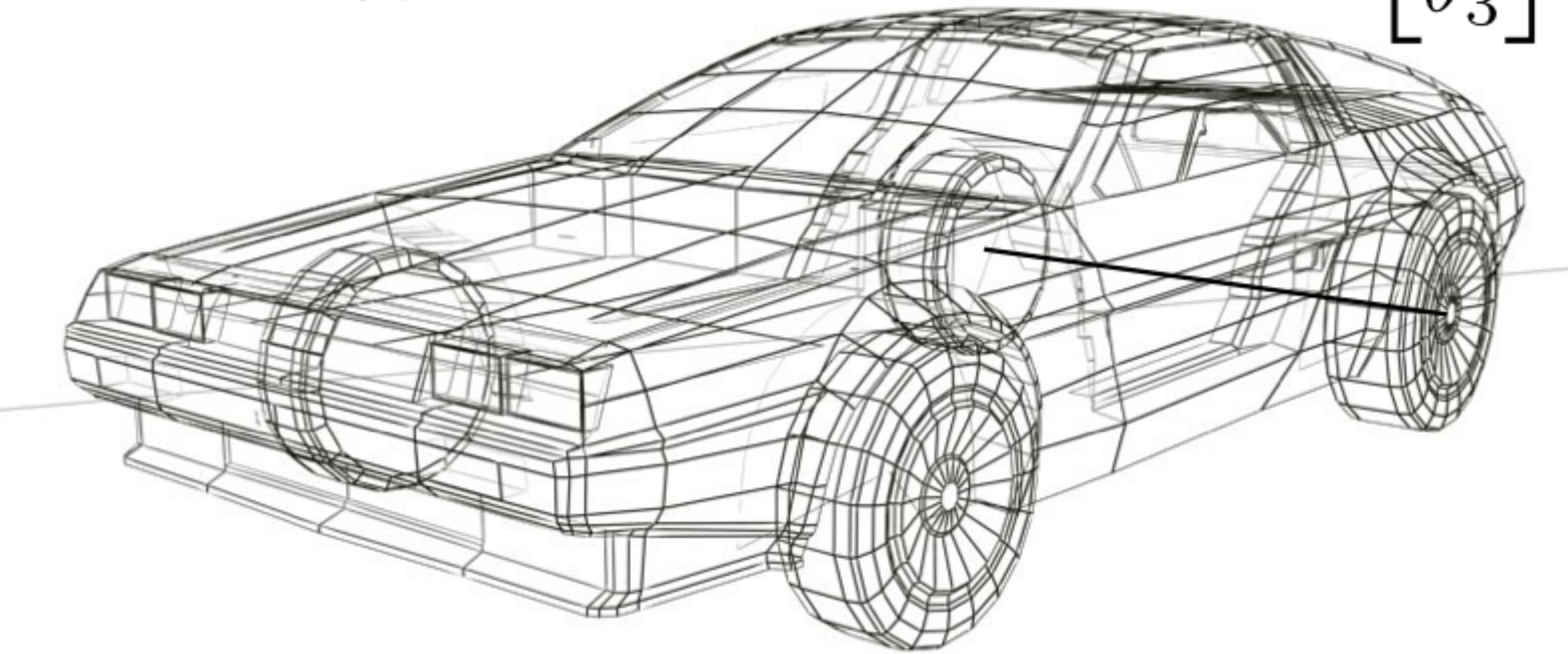
http://wiki.ros.org/tf

Goal: Reduce the amount of suffering in the world.

The Bad

- The "three number" problem
- You are given three numbers representing *the orientation of the robot*
- How many possibilities are there?

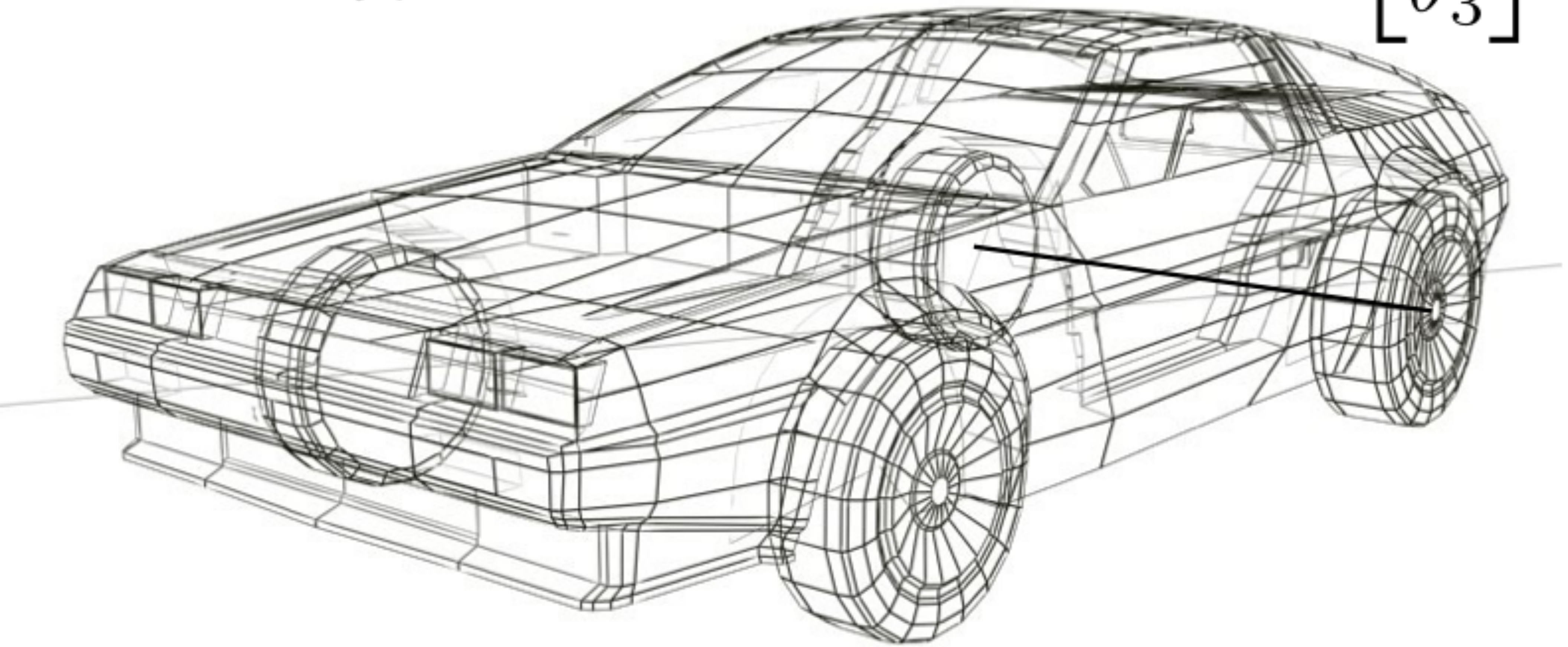
$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$



Super sweet wireframe Delorian from http://nanahra.blogspot.com

- The “three number” problem
- You are given three numbers representing *the orientation of the robot*
- How many possibilities are there?

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$

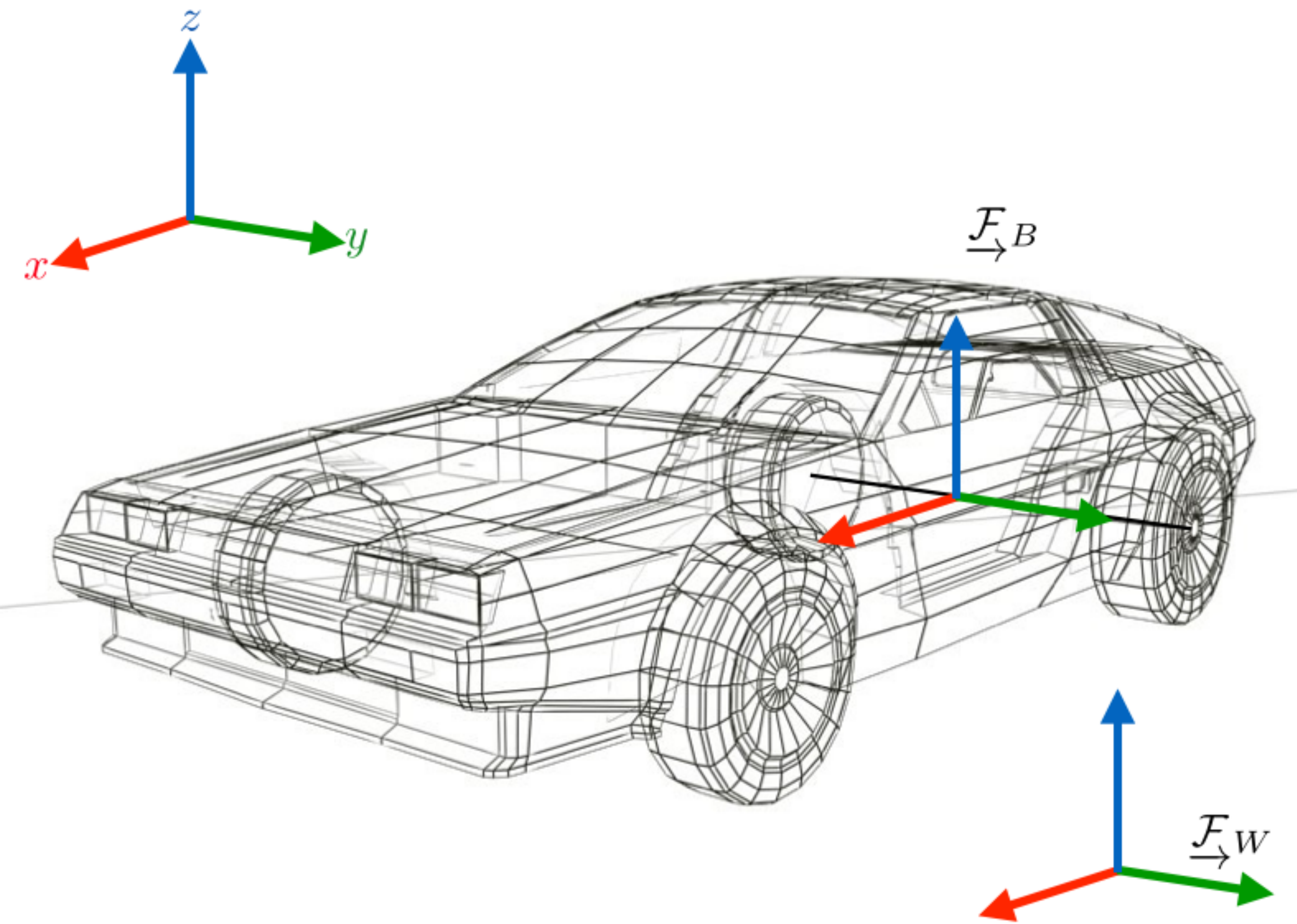


- Restrict the discussion to combinations of fundamental rotations (Euler and Tait Bryan angles)

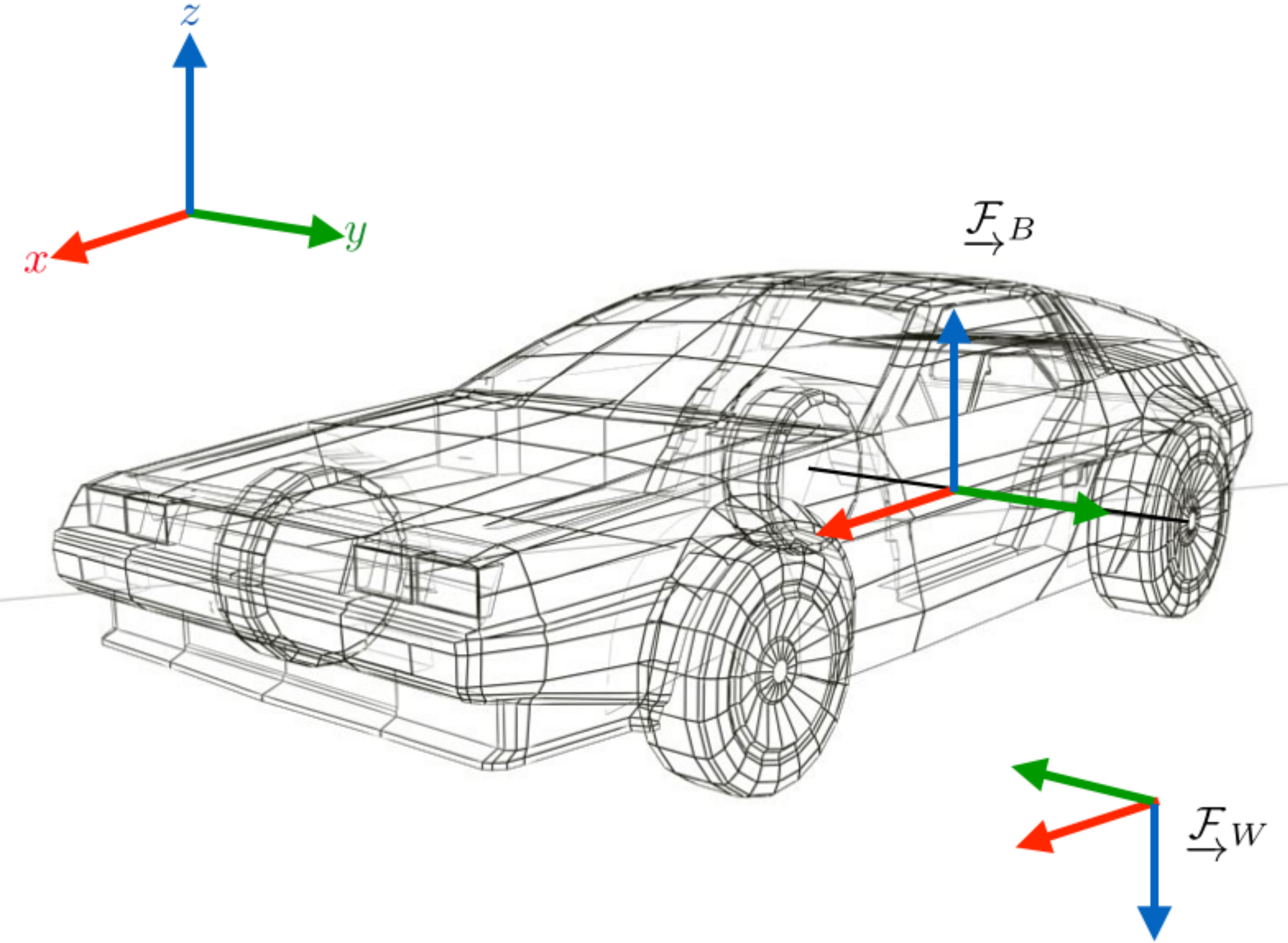
The answer is meaningless unless I provide a definition of the coordinate frames

Recommendation 1

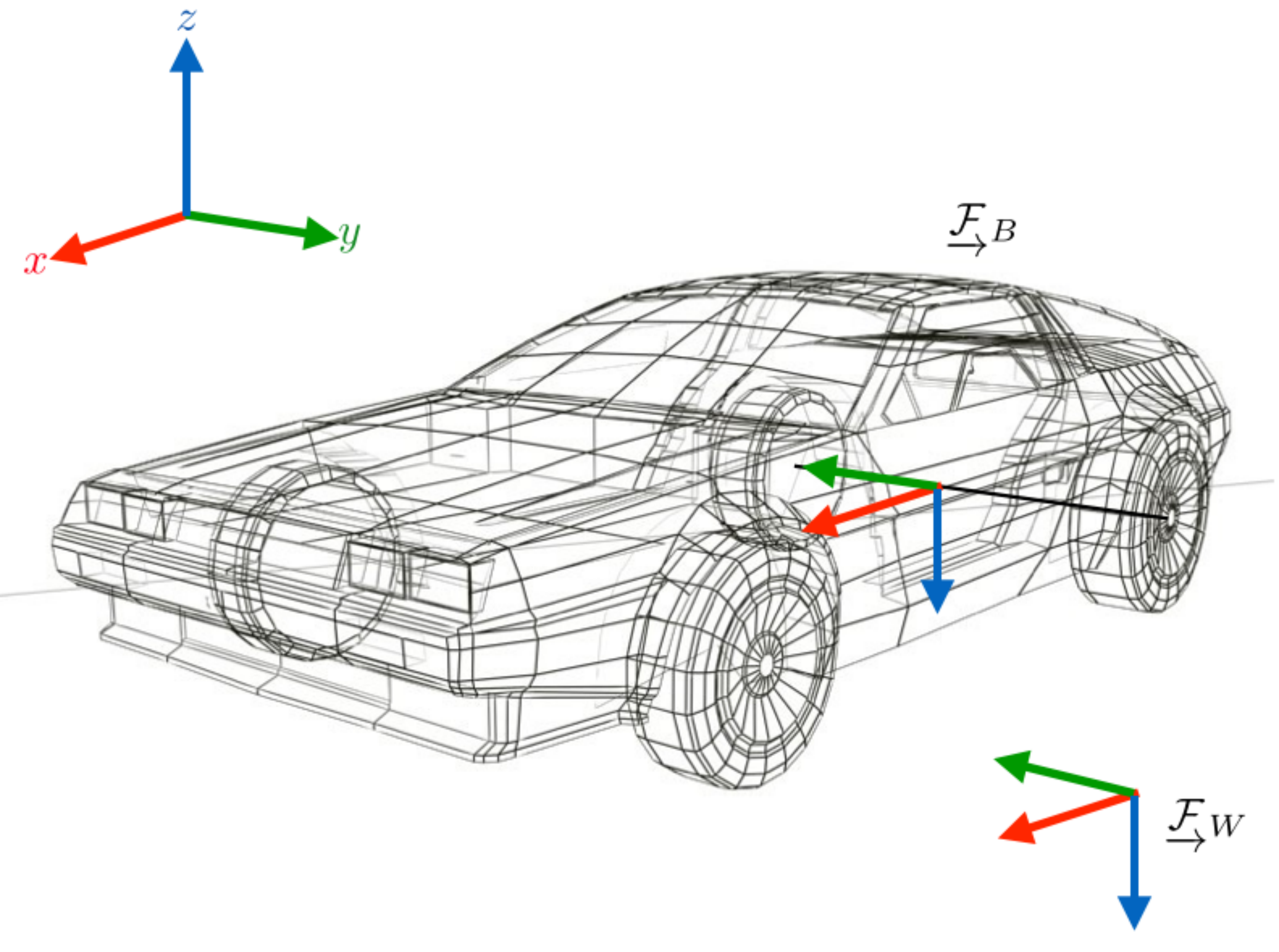
Have fear.



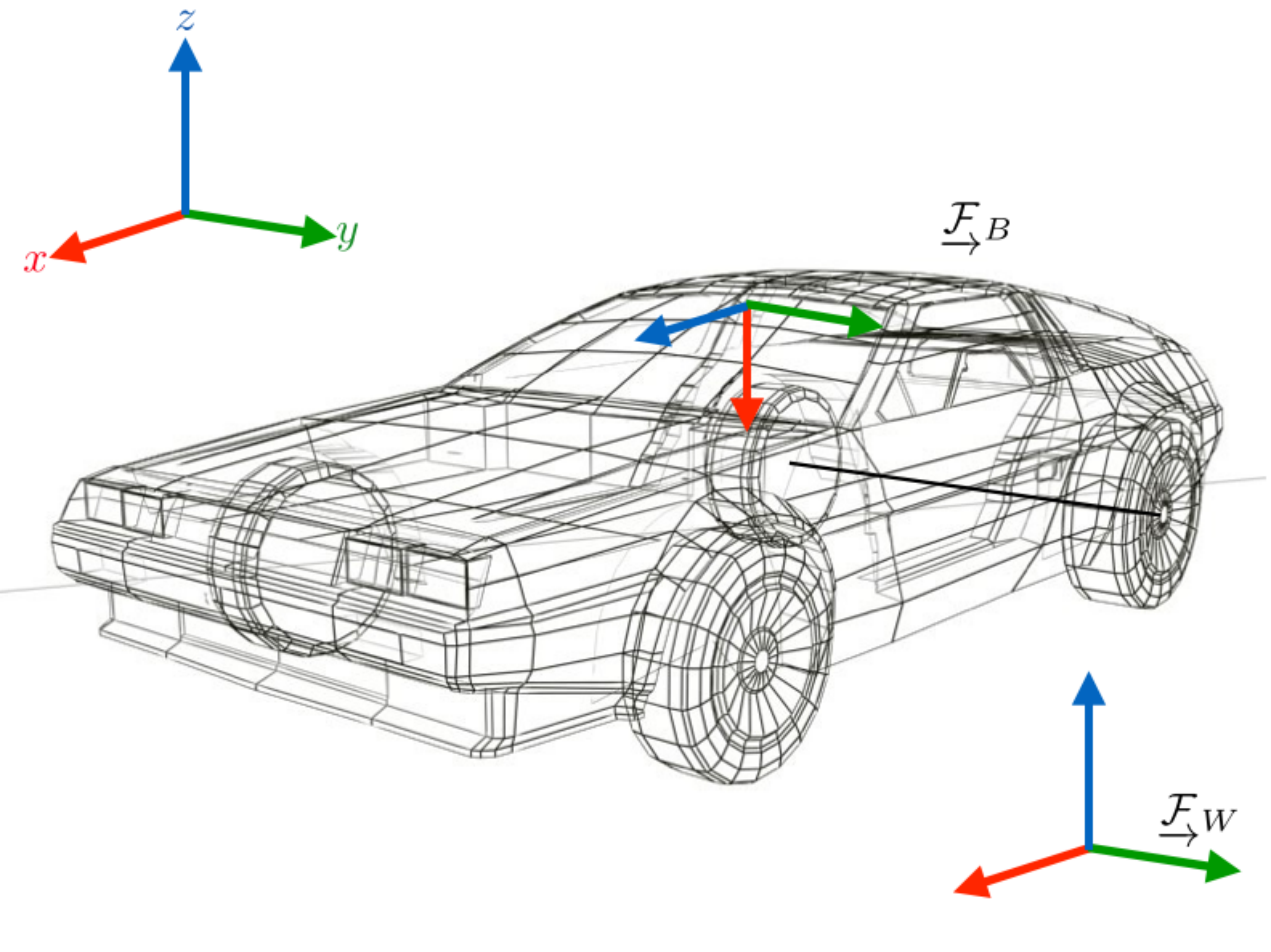
z
x
y
$\underrightarrow{\mathcal{F}}_B$
$\underrightarrow{\mathcal{F}}_W$



z
x
y
$\underset{\rightarrow}{\mathcal{F}}_B$
$\underset{\rightarrow}{\mathcal{F}}_W$



z
x
y
$\underrightarrow{\mathcal{F}}_B$
$\underrightarrow{\mathcal{F}}_W$



z
x
y
$\underrightarrow{\mathcal{F}}_B$
$\underrightarrow{\mathcal{F}}_W$

Recommendation 2

Always provide a frame diagram

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract Full Text (PDF) References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract Full Text (PDF) References

Article:
Mingyang Li and Anastasios I. Mourikis
Online temporal calibration for camera–IMU systems: Theory and algorithms
The International Journal of Robotics Research 0278364913515286, first published on May 1, 2014 as doi:10.1177/0278364913515286
Abstract Full Text (PDF) References

Article:
Marco Hutter, Hannes Sommer, Christian Gehring, Mark Hoepflinger, Michael Bloesch, and Roland Siegwart
Quadrupedal locomotion using hierarchical operational space control
The International Journal of Robotics Research 0278364913519834, first published on May 1, 2014 as doi:10.1177/0278364913519834
Abstract Full Text (PDF) References

Article:
Igor Cizelj and Calin Belta
Control of noisy differential-drive vehicles from time-bounded temporal logic specifications
The International Journal of Robotics Research 0278364914522312, first published on May 1, 2014 as doi:10.1177/0278364914522312
Abstract Full Text (PDF) References

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract Full Text (PDF) References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract Full Text (PDF) References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract Full Text (PDF) References

April 16, 2014

Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract Full Text (PDF) References

March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract Full Text (PDF)

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract | Full Text (PDF) | References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract | Full Text (PDF) | References

Article:
Mingyang Li and Anastasios I. Mourikis
Online temporal calibration for camera–IMU systems: Theory and algorithms
The International Journal of Robotics Research 0278364913515286, first published on May 1, 2014 as doi:10.1177/0278364913515286
Abstract | Full Text (PDF) | References

Article:
Marco Hutter, Hannes Sommer, Christian Gehring, Mark Hoepflinger, Michael Bloesch, and Roland Siegwart
Quadrupedal locomotion using hierarchical operational space control
The International Journal of Robotics Research 0278364913519834, first published on May 1, 2014 as doi:10.1177/0278364913519834
Abstract | Full Text (PDF) | References

Article:
Igor Cizelj and Calin Belta
Control of noisy differential-drive vehicles from time-bounded temporal logic specifications
The International Journal of Robotics Research 0278364914522312, first published on May 1, 2014 as doi:10.1177/0278364914522312
Abstract | Full Text (PDF) | References

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract | Full Text (PDF) | References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract | Full Text (PDF) | References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract | Full Text (PDF) | References

April 16, 2014

Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract | Full Text (PDF) | References

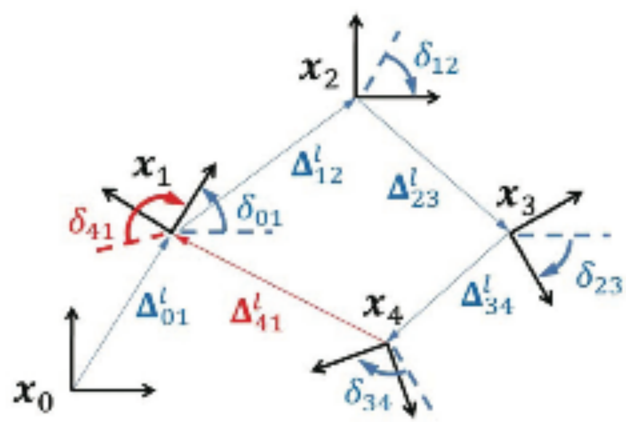
March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract | Full Text (PDF)

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract Full Text (PDF) References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract Full Text (PDF) References



$$\overline{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ \hline +1 & -1 & 0 & 0 & +1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & +1 & -1 \end{bmatrix} \quad A^{\mathsf{T}} = \begin{bmatrix} +1 & 0 & 0 & 0 \\ -1 & +1 & 0 & 0 \\ 0 & -1 & +1 & 0 \\ 0 & 0 & -1 & +1 \\ +1 & 0 & 0 & -1 \end{bmatrix}$$

A

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract Full Text (PDF) References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract Full Text (PDF) References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract Full Text (PDF) References

April 16, 2014

Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract Full Text (PDF) References

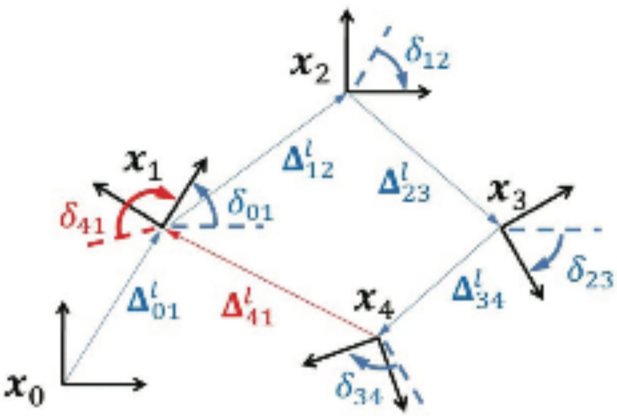
March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract Full Text (PDF)

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract Full Text (PDF) References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract Full Text (PDF) References



$$\overline{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 & +1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & +1 & -1 \end{bmatrix} \quad A^{\mathsf{T}} = \begin{bmatrix} +1 & 0 & 0 & 0 \\ -1 & +1 & 0 & 0 \\ 0 & -1 & +1 & 0 \\ 0 & 0 & -1 & +1 \\ +1 & 0 & 0 & -1 \end{bmatrix}$$

A

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract Full Text (PDF) References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract Full Text (PDF) References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract Full Text (PDF) References

April 16, 2014

Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract Full Text (PDF) References

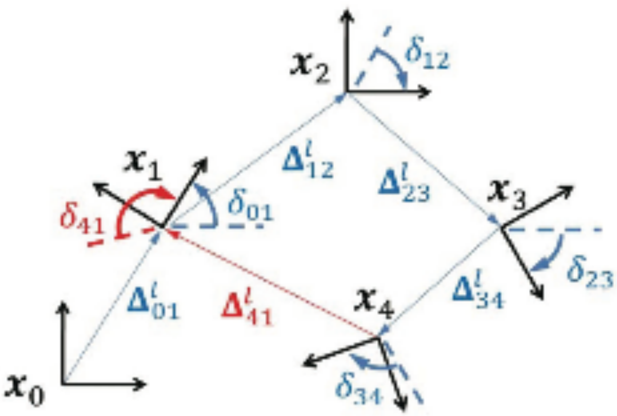
March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract Full Text (PDF)

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract Full Text (PDF) References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract Full Text (PDF) References



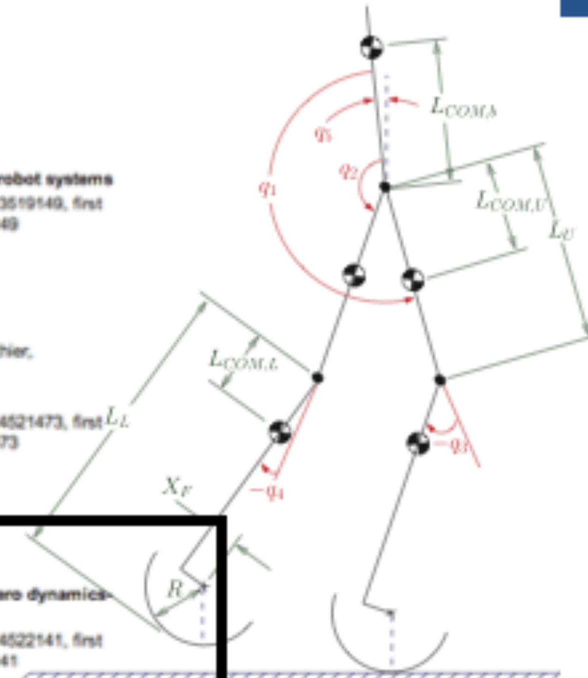
$$\overline{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ \hline +1 & -1 & 0 & 0 & +1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & +1 & -1 \end{bmatrix} \quad A^{\mathsf{T}} = \begin{bmatrix} +1 & 0 & 0 & 0 \\ -1 & +1 & 0 & 0 \\ 0 & -1 & +1 & 0 \\ 0 & 0 & -1 & +1 \\ +1 & 0 & 0 & -1 \end{bmatrix}$$

A

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract Full Text (PDF) References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract Full Text (PDF) References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract Full Text (PDF) References



April 16, 2014

Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract Full Text (PDF) References

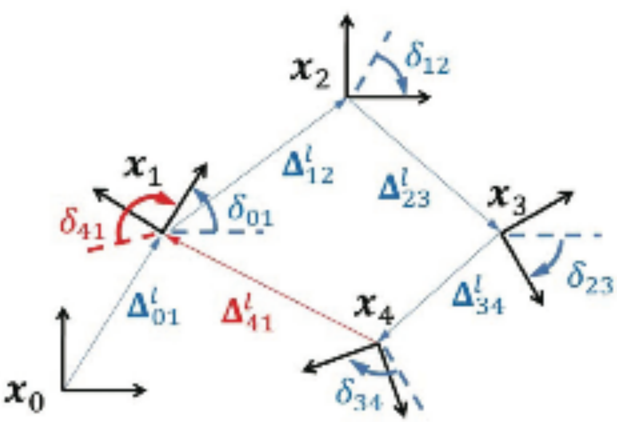
March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract Full Text (PDF)

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract Full Text (PDF) References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract Full Text (PDF) References



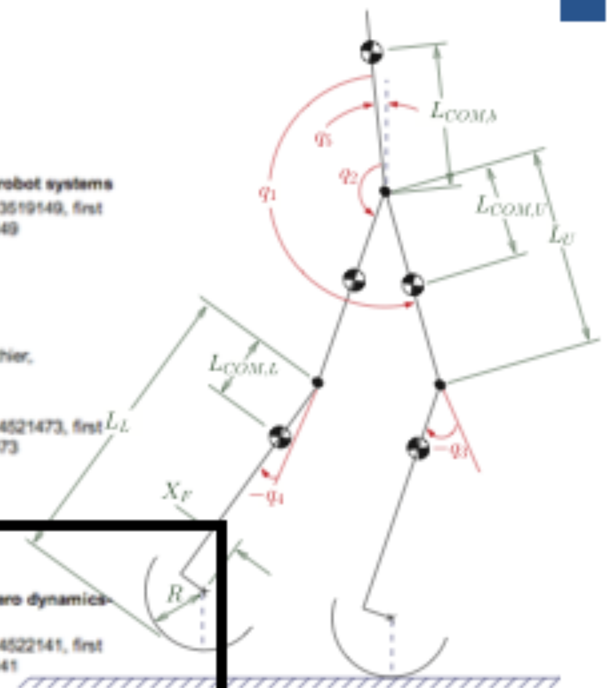
$$\overline{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 & +1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & +1 & -1 \end{bmatrix} \qquad A^{\mathsf{T}} = \begin{bmatrix} +1 & 0 & 0 & 0 \\ -1 & +1 & 0 & 0 \\ 0 & -1 & +1 & 0 \\ 0 & 0 & -1 & +1 \\ +1 & 0 & 0 & -1 \end{bmatrix}$$

A

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract Full Text (PDF) References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract Full Text (PDF) References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract Full Text (PDF) References



April 16, 2014

Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract Full Text (PDF) References

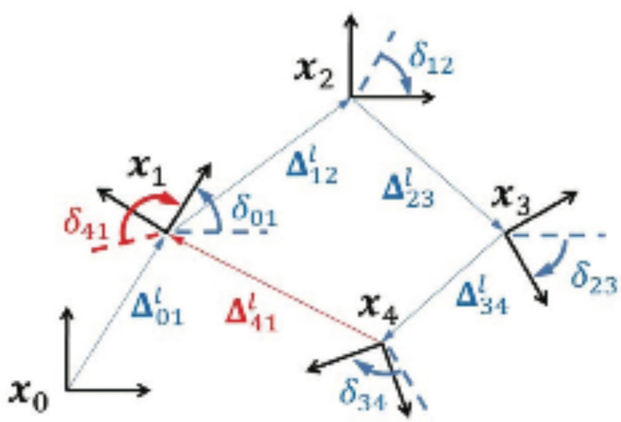
March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract Full Text (PDF)

May 1, 2014

Article:
Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber
Hierarchical quadratic programming: Fast online humanoid-robot motion generation
The International Journal of Robotics Research 0278364914521306, first published on May 1, 2014 as doi:10.1177/0278364914521306
Abstract | Full Text (PDF) | References

Article:
Luca Carlone, Rosario Aragues, José A. Castellanos, and Basilio Bona
A fast and accurate approximation for planar pose graph optimization
The International Journal of Robotics Research 0278364914523689, first published on May 1, 2014 as doi:10.1177/0278364914523689
Abstract | Full Text (PDF) | References



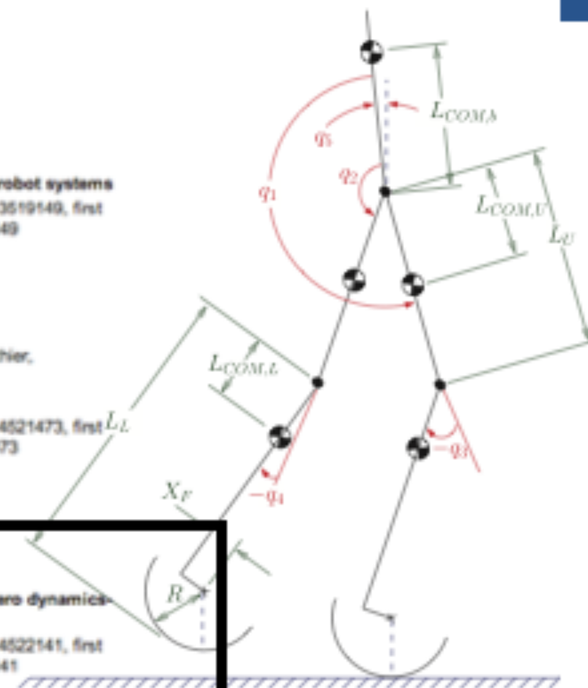
$$\overline{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 & +1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & +1 & -1 \end{bmatrix} \qquad A^{\mathsf{T}} = \begin{bmatrix} +1 & 0 & 0 & 0 \\ -1 & +1 & 0 & 0 \\ 0 & -1 & +1 & 0 \\ 0 & 0 & -1 & +1 \\ +1 & 0 & 0 & -1 \end{bmatrix}$$

A

Article:
Gianluca Antonelli, Filippo Arrichiello, Fabrizio Caccavale, and Alessandro Marino
Decentralized time-varying formation control for multi-robot systems
The International Journal of Robotics Research 0278364913519149, first published on May 1, 2014 as doi:10.1177/0278364913519149
Abstract | Full Text (PDF) | References

Article:
Andrew T. Baisch, Onur Ozcan, Benjamin Goldberg, Daniel Ithier, and Robert J. Wood
High speed locomotion for a quadrupedal microrobot
The International Journal of Robotics Research 0278364914521473, first published on May 1, 2014 as doi:10.1177/0278364914521473
Abstract | Full Text (PDF) | References

Article:
Anne E. Martin, David C. Post, and James P. Schmiedeler
Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet
The International Journal of Robotics Research 0278364914522141, first published on May 1, 2014 as doi:10.1177/0278364914522141
Abstract | Full Text (PDF) | References

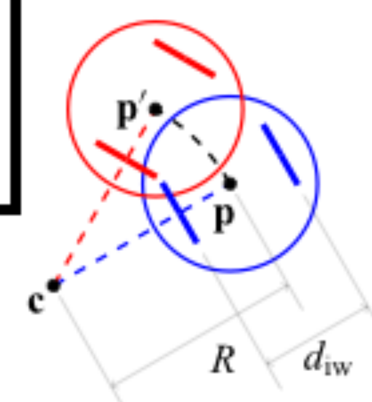


April 16, 2014

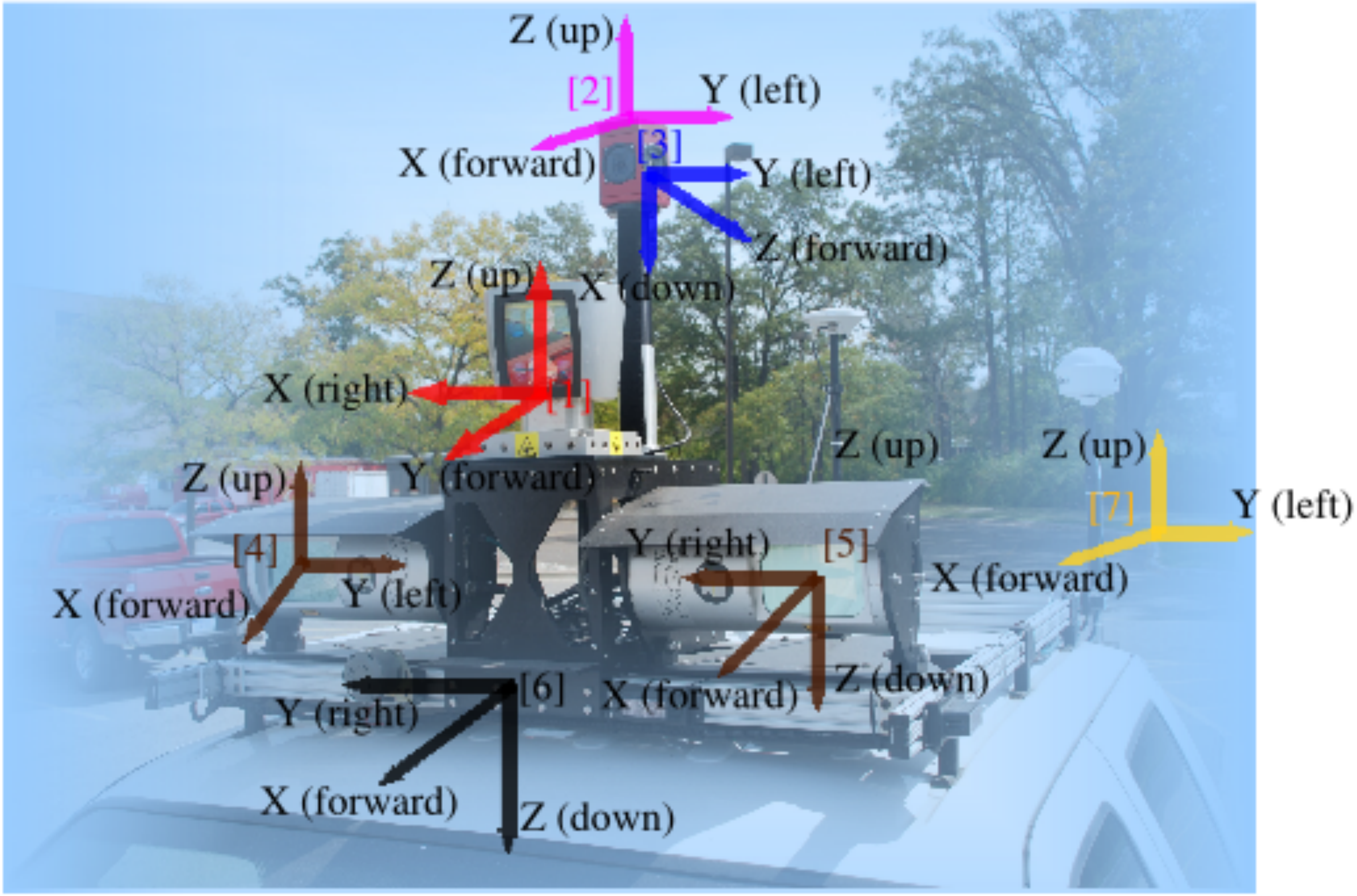
Article:
Melvin Gauci, Jianing Chen, Wei Li, Tony J. Dodd, and Roderich Groß
Self-organized aggregation without computation
The International Journal of Robotics Research 0278364914525244, first published on April 16, 2014 as doi:10.1177/0278364914525244
Abstract | Full Text (PDF) | References

March 14, 2011

T William Mather and M Ani Hsieh
Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles
The International Journal of Robotics Research 0278364911401442, first published on March 14, 2011 as doi:10.1177/0278364911401442
Abstract | Full Text (PDF)



The Bad

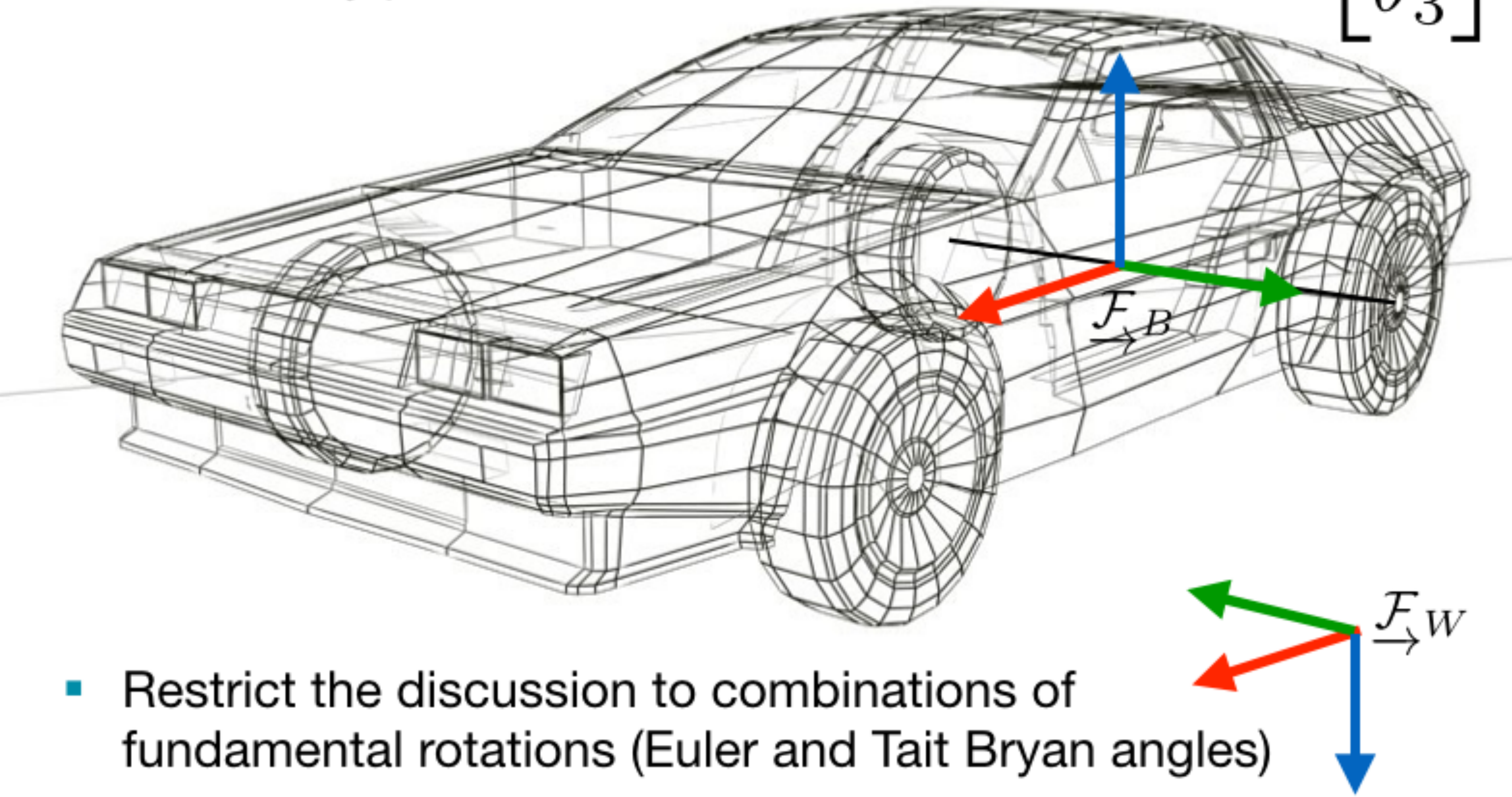


[1] Velodyne, [2] Ladybug3 (actual location: center of camera system),
[3] Ladybug3 Camera 5, [4] Right Riegl, [5] Left Riegl,
[6] Body Frame (actual location: center of rear axle)
[7] Local Frame (Angle between the X-axis and East is known)

Gaurav Pandey, James R. McBride and Ryan M. Eustice, Ford campus vision and lidar data set. International Journal of Robotics Research, 30(13):1543-1552, November 2011.

- The “three number” problem
- You are given three numbers representing *the orientation of the robot*
- How many possibilities are there?

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$



- Restrict the discussion to combinations of fundamental rotations (Euler and Tait Bryan angles)

The Bad: Choices

- There are 12 different valid combinations of fundamental rotations
  - z-x-z, x-y-x, y-z-y, z-y-z, x-z-x, y-x-y
  - x-y-z, y-z-x, z-y-x, x-z-y, z-y-x, y-x-z

The Bad: Choices

- There are 12 different valid combinations of fundamental rotations
  - z-x-z, x-y-x, y-z-y, z-y-z, x-z-x, y-x-y
  - x-y-z, y-z-x, z-y-x, x-z-y, z-y-x, y-x-z
- But, we can build fundamental rotations in two forms, active or passive

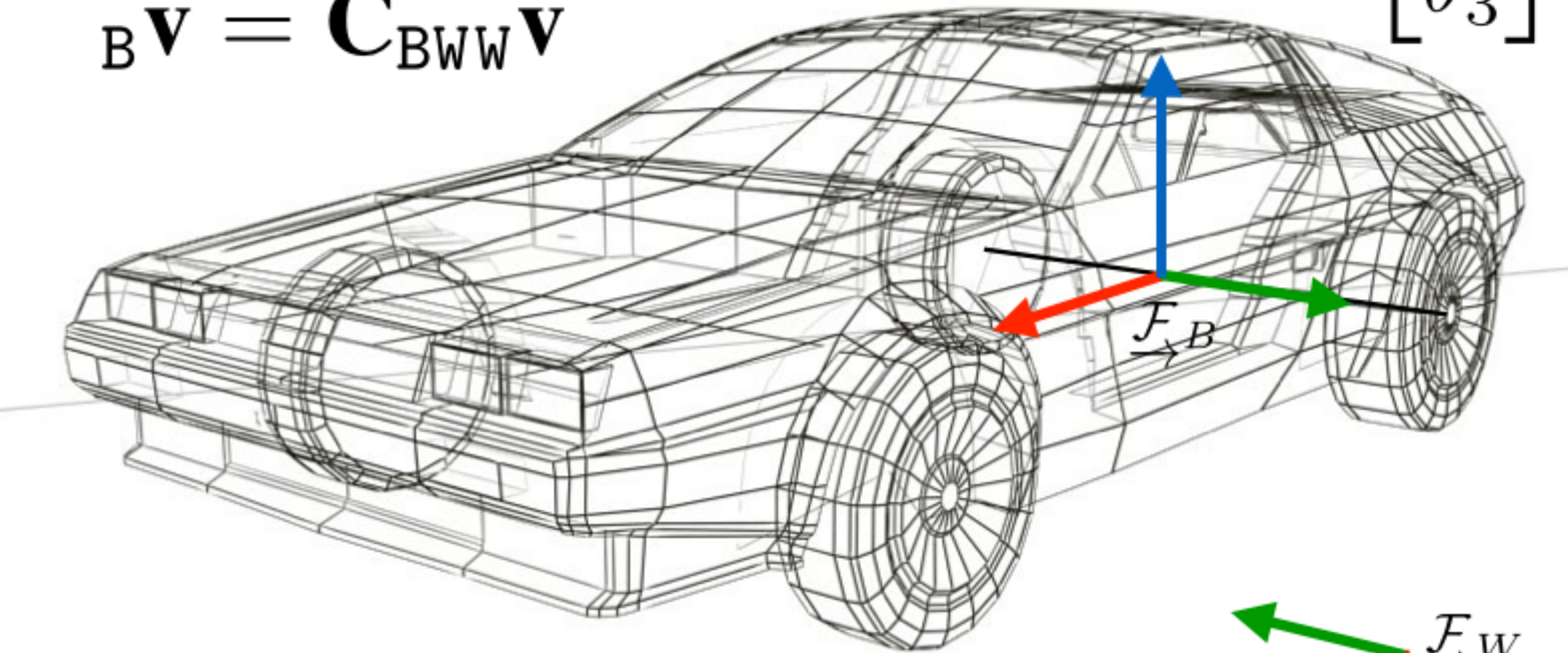
$$\mathbf{R}_z(\alpha) := \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{C}_z(\alpha) := \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

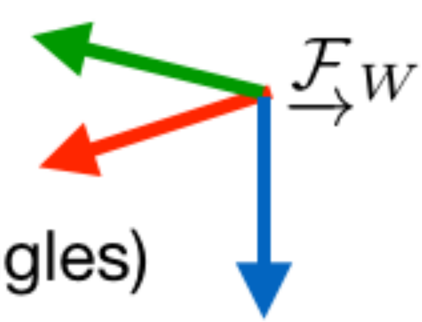
- Which rotation matrix are we talking about?

$$ {}_{\mathtt{W}}\mathbf{v} = \mathbf{C}_{\mathtt{WB}}\,{}_{\mathtt{B}}\mathbf{v} $$

$$ {}_{\mathtt{B}}\mathbf{v} = \mathbf{C}_{\mathtt{BW}}\,{}_{\mathtt{W}}\mathbf{v} $$

$$ \mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} $$





- Restrict the discussion to combinations of fundamental rotations (Euler and Tait Bryan angles)

The Bad: Choices

- There are 12 different valid combinations of fundamental rotations
- active or passive
- body-to-world or world-to-body

The Bad: Choices

- There are 12 different valid combinations of fundamental rotations
- active or passive
- body-to-world or world-to-body
- Ordering of parameters
  - 1-2-3, 1-3-2, 2-1-3, 2-3-1, 3-1-2, 3-2-1

The Bad: Choices

- There are 12 different valid combinations of fundamental rotations
- active or passive
- body-to-world or world-to-body
- Ordering of parameters
  - 1-2-3, 1-3-2, 2-1-3, 2-3-1, 3-1-2, 3-2-1
- Degrees or radians

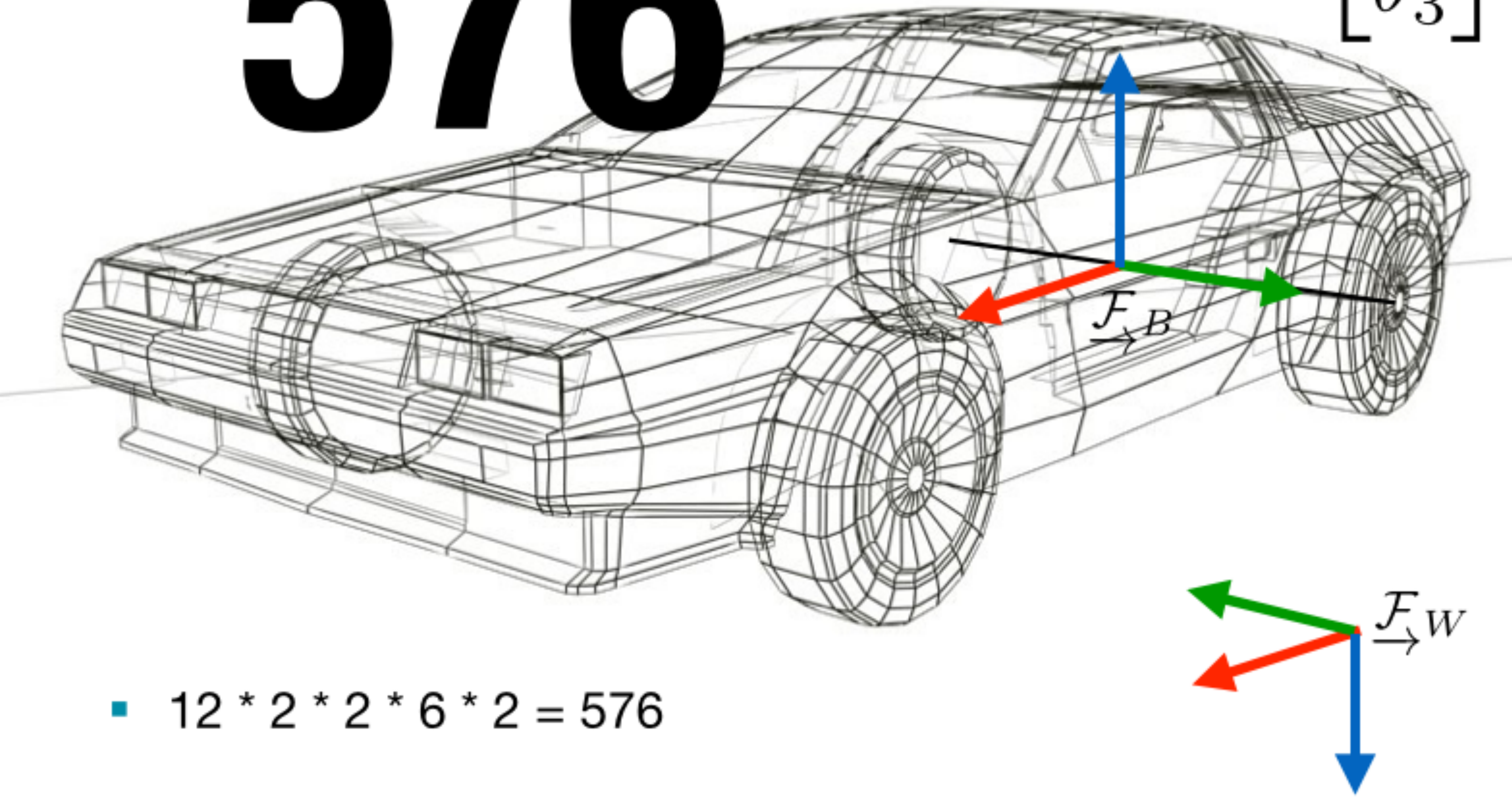
The Bad: Choices

- (12) There are 12 different valid combinations of fundamental rotations
- (2) active or passive
- (2) body-to-world or world-to-body
- (6) Ordering of parameters
  - 1-2-3, 1-3-2, 2-1-3, 2-3-1, 3-1-2, 3-2-1
- (2) Degrees or radians

- The “three number” problem
- How many possibilities are there?

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$

576

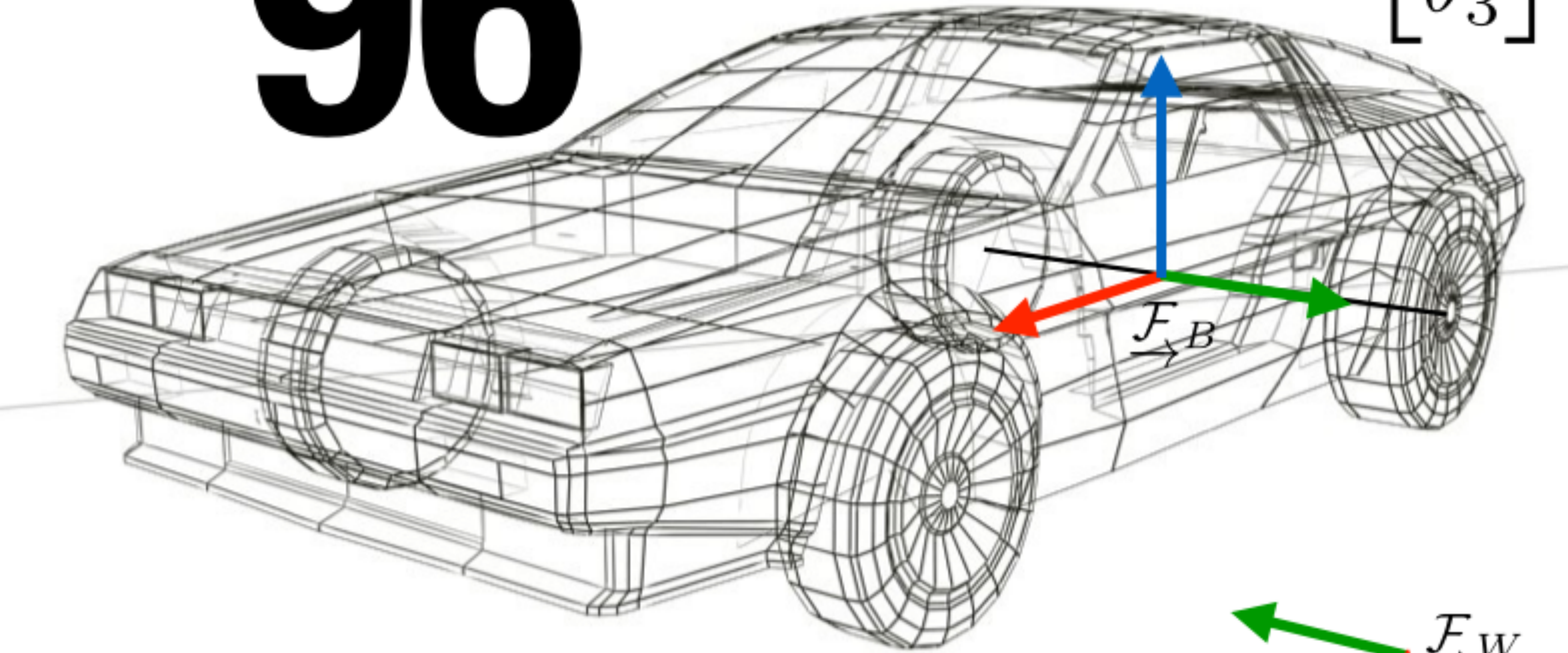


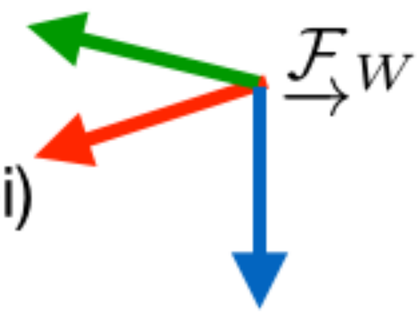
- 12 * 2 * 2 * 6 * 2 = 576

- The “three number” problem
- How many possibilities are there?

96

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$

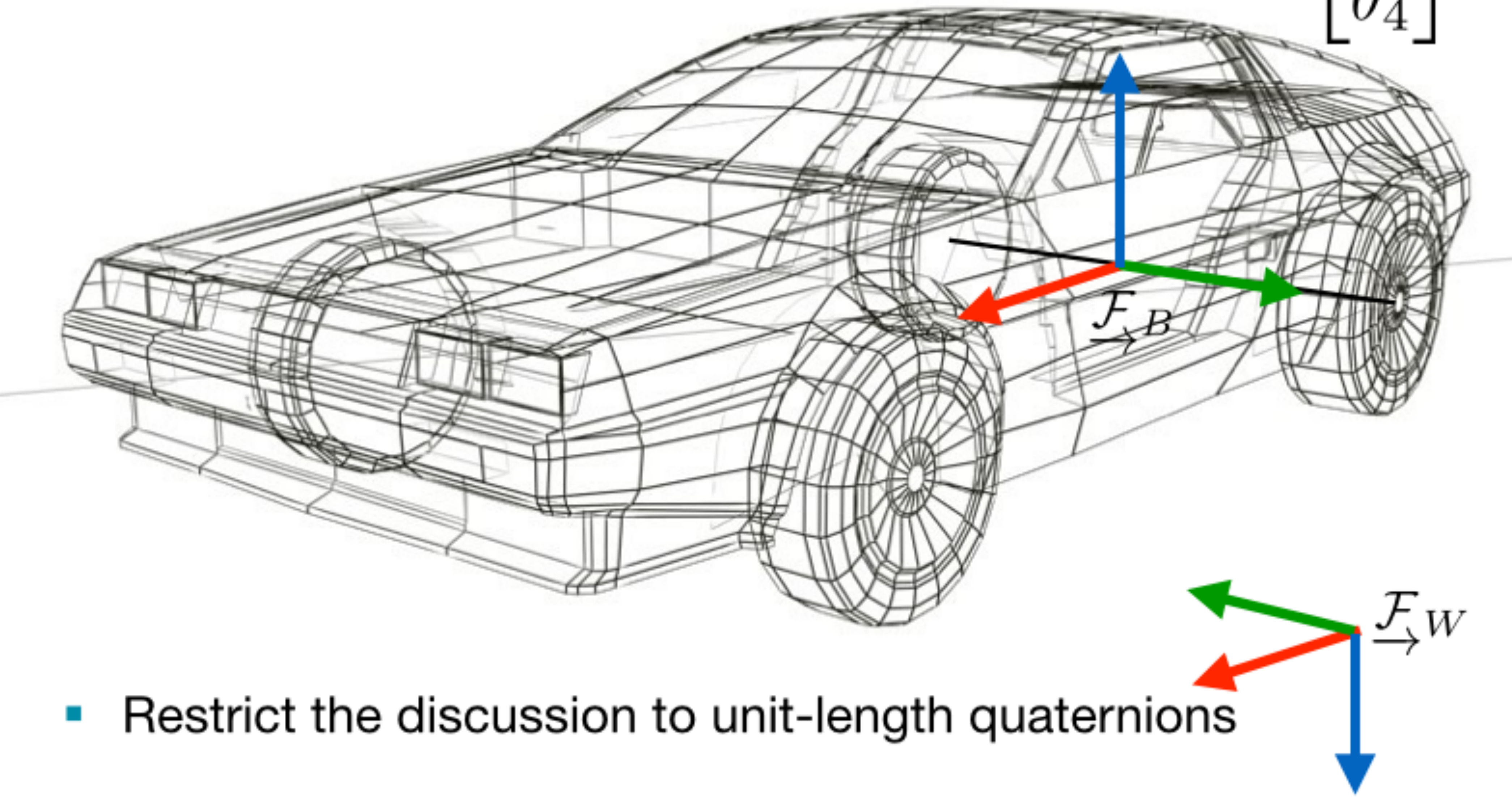




- If you know that the sequence is i-j-k (or k-j-i)

- The “four number” problem
- How many possibilities are there?

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix}$$



- Restrict the discussion to unit-length quaternions

The Bad: Quaternions

- Even with unit-length quaternions, there are choices

The Bad: Quaternions

- Even with unit-length quaternions, there are choices
- Parameter ordering
  - We won't consider arbitrary ordering
  - We do have to decide on scalar first or scalar last

The Bad: Quaternions

- Even with unit-length quaternions, there are choices
- Parameter ordering
  - We won’t consider arbitrary ordering
  - We do have to decide on scalar first or scalar last

$$Q = w + xi + yj + zk$$

The Bad: Quaternions

- Even with unit-length quaternions, there are choices
- Parameter ordering
  - We won't consider arbitrary ordering
  - We do have to decide on scalar first or scalar last

$$Q = w + xi + yj + zk$$

$$\mathbf{q} := \begin{bmatrix} w \\ x \\ y \\ z \end{bmatrix}$$ Scalar First

The Bad: Quaternions

- Even with unit-length quaternions, there are choices
- Parameter ordering
  - We won't consider arbitrary ordering
  - We do have to decide on scalar first or scalar last

$$Q = w + xi + yj + zk$$

$$\mathbf{q} := \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Scalar Last

SEARCH RESULTS

You searched for: **((quaternion) AND robot)**

You Refined by:

Publisher: IEEE

Content Type: Journals & Magazines

47 Results returned

Results per page 25

Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB

Corke, P.

Robotics & Automation Magazine, IEEE

Volume: 3 , Issue: 1

Digital Object Identifier: 10.1109/100.486658

Publication Year: 1996 , Page(s): 24 - 32

Cited by: Papers (222)

IEEE JOURNALS & MAGAZINES

| Quick Abstract | PDF (876 KB)

The attitude control problem

Wen, J.T.-Y. ; Kreutz-Delgado, K.

Automatic Control, IEEE Transactions on

Volume: 36 , Issue: 10

Digital Object Identifier: 10.1109/9.90228

Publication Year: 1991 , Page(s): 1148 - 1162

Cited by: Papers (195) | Patents (1)

IEEE JOURNALS & MAGAZINES

| Quick Abstract | PDF (1240 KB)

Attitude control without angular velocity measurement: a passivity approach

Lizarralde, F. ; Wen, J.T.

Automatic Control, IEEE Transactions on

Volume: 41 , Issue: 3

Digital Object Identifier: 10.1109/9.486654

Publication Year: 1996 , Page(s): 468 - 472

Cited by: Papers (118)

IEEE JOURNALS & MAGAZINES

| Quick Abstract | PDF (404 KB)

Attitude stabilization of a VTOL quadrotor aircraft

Tayebi, A. ; McGilvray, S.

Control Systems Technology, IEEE Transactions on

Volume: 14 , Issue: 3

Digital Object Identifier: 10.1109/TCST.2006.872519

Publication Year: 2006 , Page(s): 562 - 571

Cited by: Papers (109) | Patents (1)

IEEE JOURNALS & MAGAZINES

| Quick Abstract | PDF (712 KB) | HTML

SEARCH RESULTS

You searched for: ((quaternion) AND robot)
You Refined by:
Publisher: IEEE
Content Type: Journals & Magazines

47 Results returned

Results per page 25

Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

Scalar first

SEARCH RESULTS

You searched for: **((quaternion) AND robot)**

You Refined by:

Publisher: IEEE

Content Type: Journals & Magazines

47 Results returned

Results per page 25 **Sort by:** Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
| Quick Abstract | PDF (876 KB)

Scalar first

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
| Quick Abstract | PDF (1240 KB)

Scalar first

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
| Quick Abstract | PDF (404 KB)

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
| Quick Abstract | PDF (712 KB) | HTML

SEARCH RESULTS

You searched for: **((quaternion) AND robot)**

You Refined by:

Publisher: IEEE

Content Type: Journals & Magazines

47 Results returned

Results per page 25 Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

Scalar first

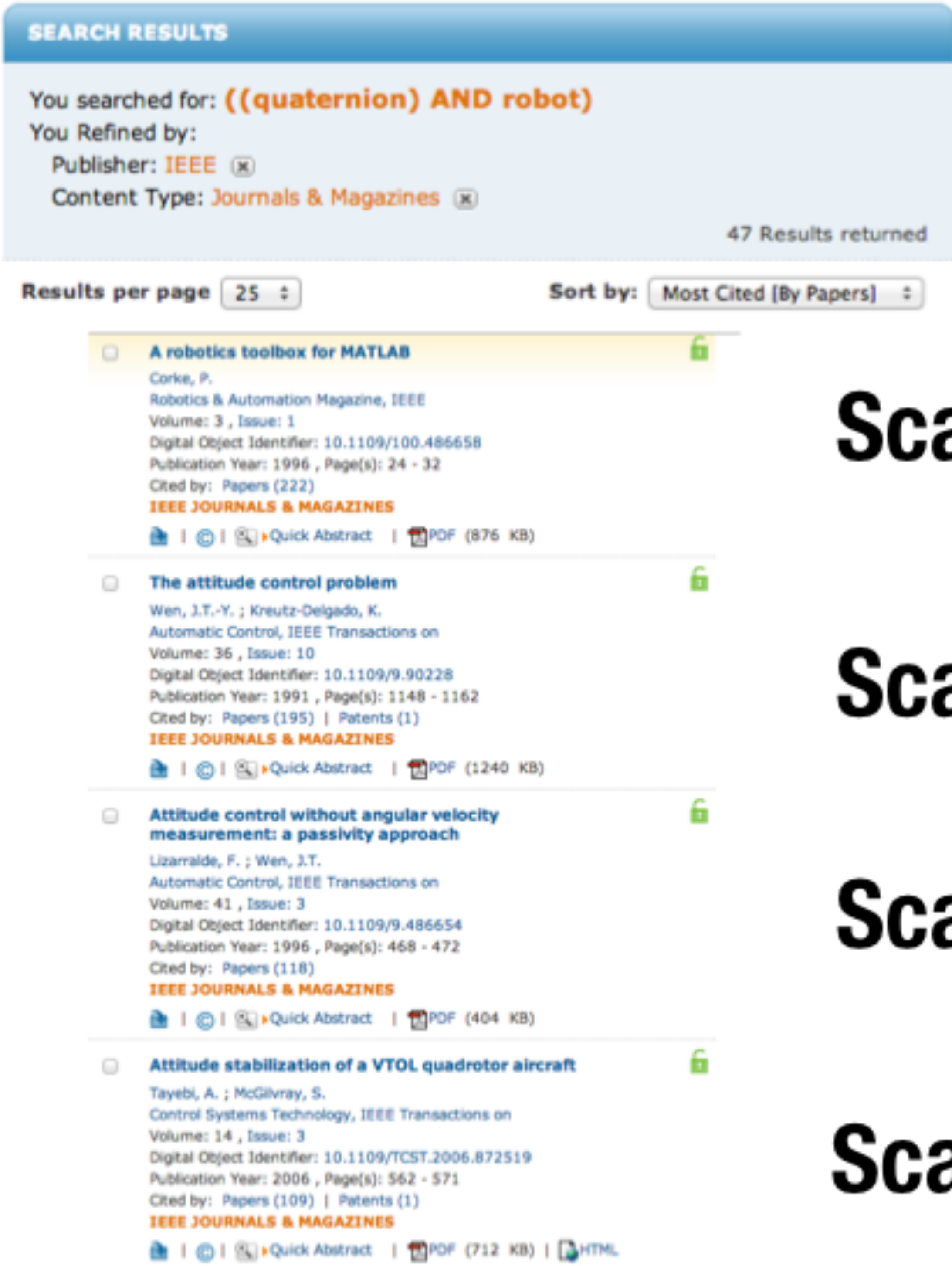
The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Scalar first

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Scalar first

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML



SEARCH RESULTS

You searched for: ((quaternion) AND robot)
You Refined by:
Publisher: IEEE
Content Type: Journals & Magazines

47 Results returned

Results per page 25

Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

Scalar first

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Scalar first

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Scalar first

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

Scalar last

The Bad: Quaternions

- Even with unit-length quaternions, there are choices
  Parameter ordering
  - We won't consider arbitrary ordering
  - We do have to decide on scalar first or scalar last
- What is quaternion multiplication (tells you how to build a rotation matrix)
  - Hamilton or JPL

The Bad: Quaternions

$$Q = q_0 + q_1 i + q_2 j + q_3 k$$

- Hamilton

$$i^2 = j^2 = k^2 = ijk = -1$$

$$ij = k, \; jk = i, \; ki = j$$

The Bad: Quaternions

$$Q = q_0 + q_1 i + q_2 j + q_3 k$$

- Hamilton

$$i^2 = j^2 = k^2 = ijk = -1$$
$$ij = k,\ jk = i,\ ki = j$$

- JPL/Shuster

$$i^2 = j^2 = k^2 = -1,\ ijk = 1$$
$$ij = -k,\ jk = -i,\ ki = -j$$

$$\mathbf{C}_{\mathrm{H}}(Q) = \mathbf{C}_{\mathrm{J}}(Q)^T$$

SEARCH RESULTS

You searched for: **((quaternion) AND robot)**

You Refined by:

Publisher: IEEE

Content Type: Journals & Magazines

47 Results returned

Results per page: 25

Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

SEARCH RESULTS

You searched for: ((quaternion) AND robot)
You Refined by:
Publisher: IEEE
Content Type: Journals & Magazines

47 Results returned

Results per page 25 Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

Not stated

SEARCH RESULTS

You searched for: **((quaternion) AND robot)**
You Refined by:
Publisher: IEEE
Content Type: Journals & Magazines

47 Results returned

Results per page 25 **Sort by:** Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

Not stated

Hamilton

SEARCH RESULTS

You searched for: **((quaternion) AND robot)**
You Refined by:
Publisher: IEEE
Content Type: Journals & Magazines

47 Results returned

Results per page 25 **Sort by:** Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

Not stated

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Hamilton

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Hamilton

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

SEARCH RESULTS

You searched for: ((quaternion) AND robot)
You Refined by:
Publisher: IEEE
Content Type: Journals & Magazines

47 Results returned

Results per page 25 Sort by: Most Cited [By Papers]

A robotics toolbox for MATLAB
Corke, P.
Robotics & Automation Magazine, IEEE
Volume: 3 , Issue: 1
Digital Object Identifier: 10.1109/100.486658
Publication Year: 1996 , Page(s): 24 - 32
Cited by: Papers (222)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (876 KB)

Not stated

The attitude control problem
Wen, J.T.-Y. ; Kreutz-Delgado, K.
Automatic Control, IEEE Transactions on
Volume: 36 , Issue: 10
Digital Object Identifier: 10.1109/9.90228
Publication Year: 1991 , Page(s): 1148 - 1162
Cited by: Papers (195) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (1240 KB)

Hamilton

Attitude control without angular velocity measurement: a passivity approach
Lizarralde, F. ; Wen, J.T.
Automatic Control, IEEE Transactions on
Volume: 41 , Issue: 3
Digital Object Identifier: 10.1109/9.486654
Publication Year: 1996 , Page(s): 468 - 472
Cited by: Papers (118)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (404 KB)

Hamilton

Attitude stabilization of a VTOL quadrotor aircraft
Tayebi, A. ; McGilvray, S.
Control Systems Technology, IEEE Transactions on
Volume: 14 , Issue: 3
Digital Object Identifier: 10.1109/TCST.2006.872519
Publication Year: 2006 , Page(s): 562 - 571
Cited by: Papers (109) | Patents (1)
IEEE JOURNALS & MAGAZINES
Quick Abstract | PDF (712 KB) | HTML

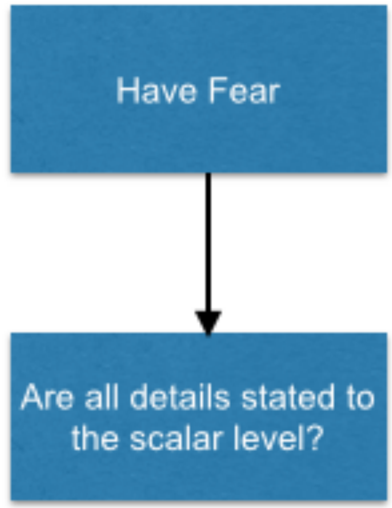
Not stated

Quaternion decision tree

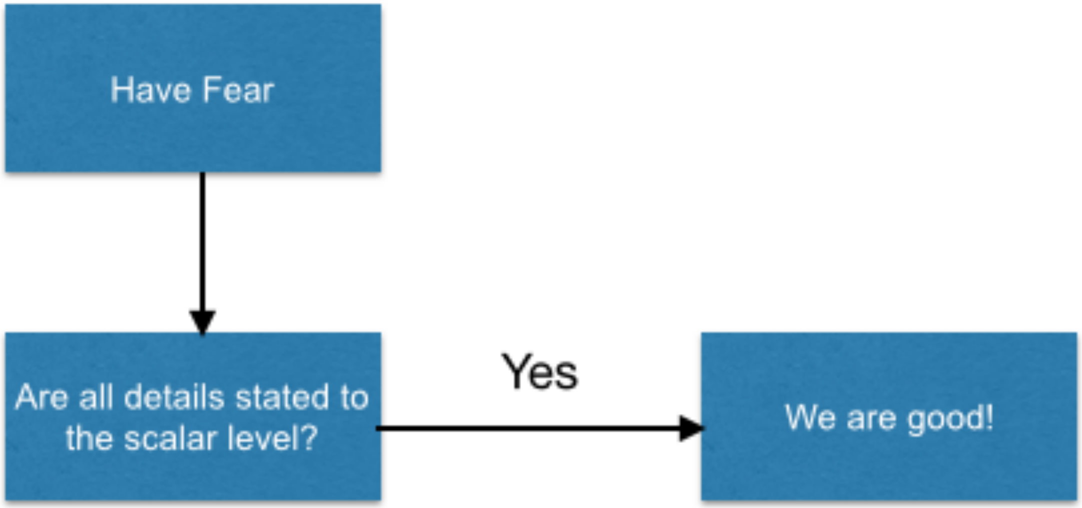
Quaternion decision tree

Have Fear

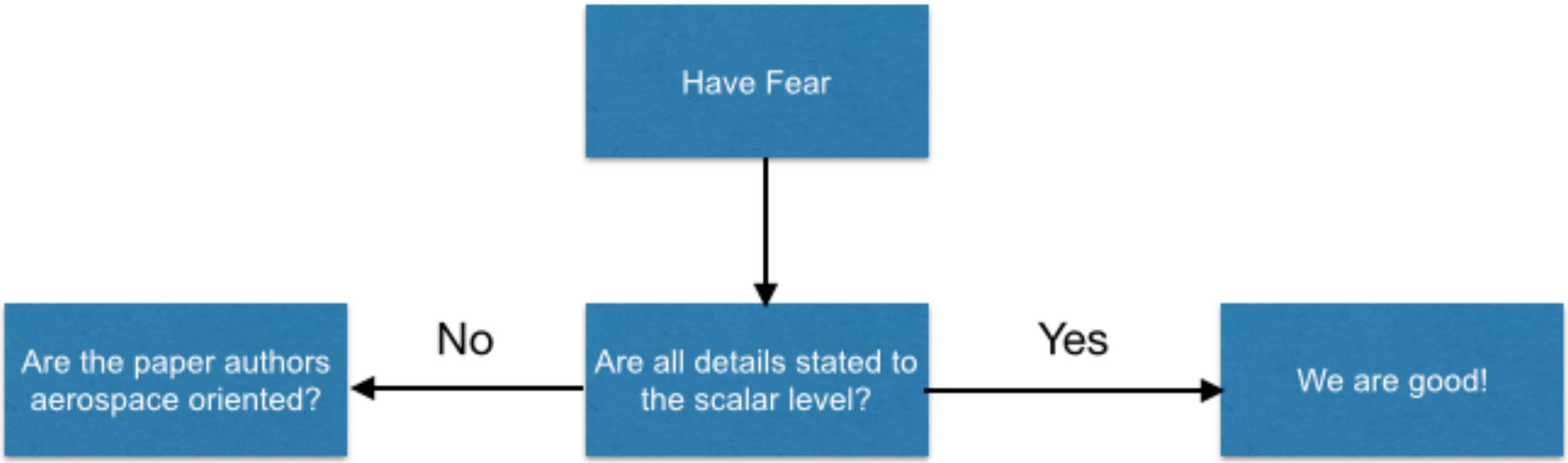
Quaternion decision tree



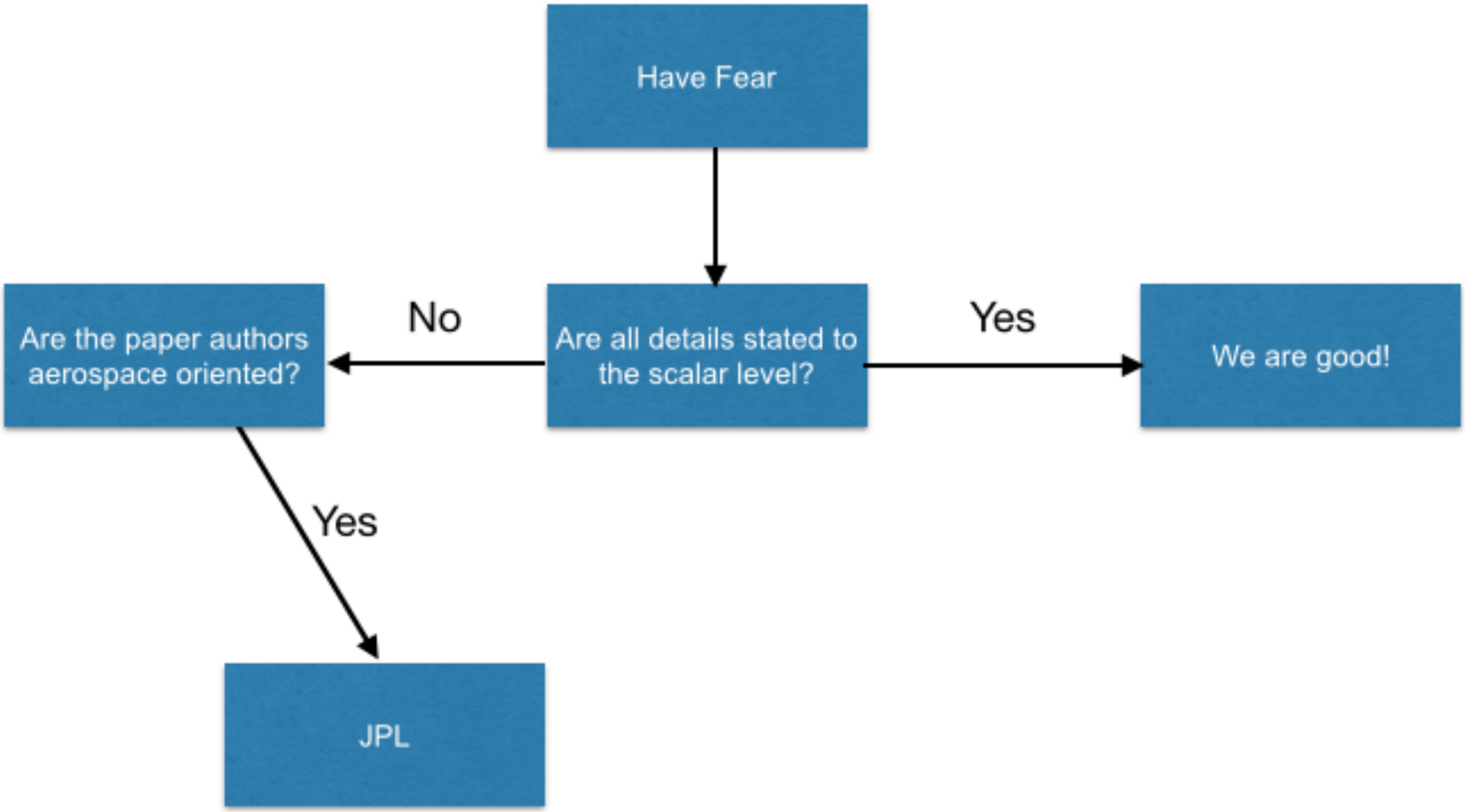
Quaternion decision tree



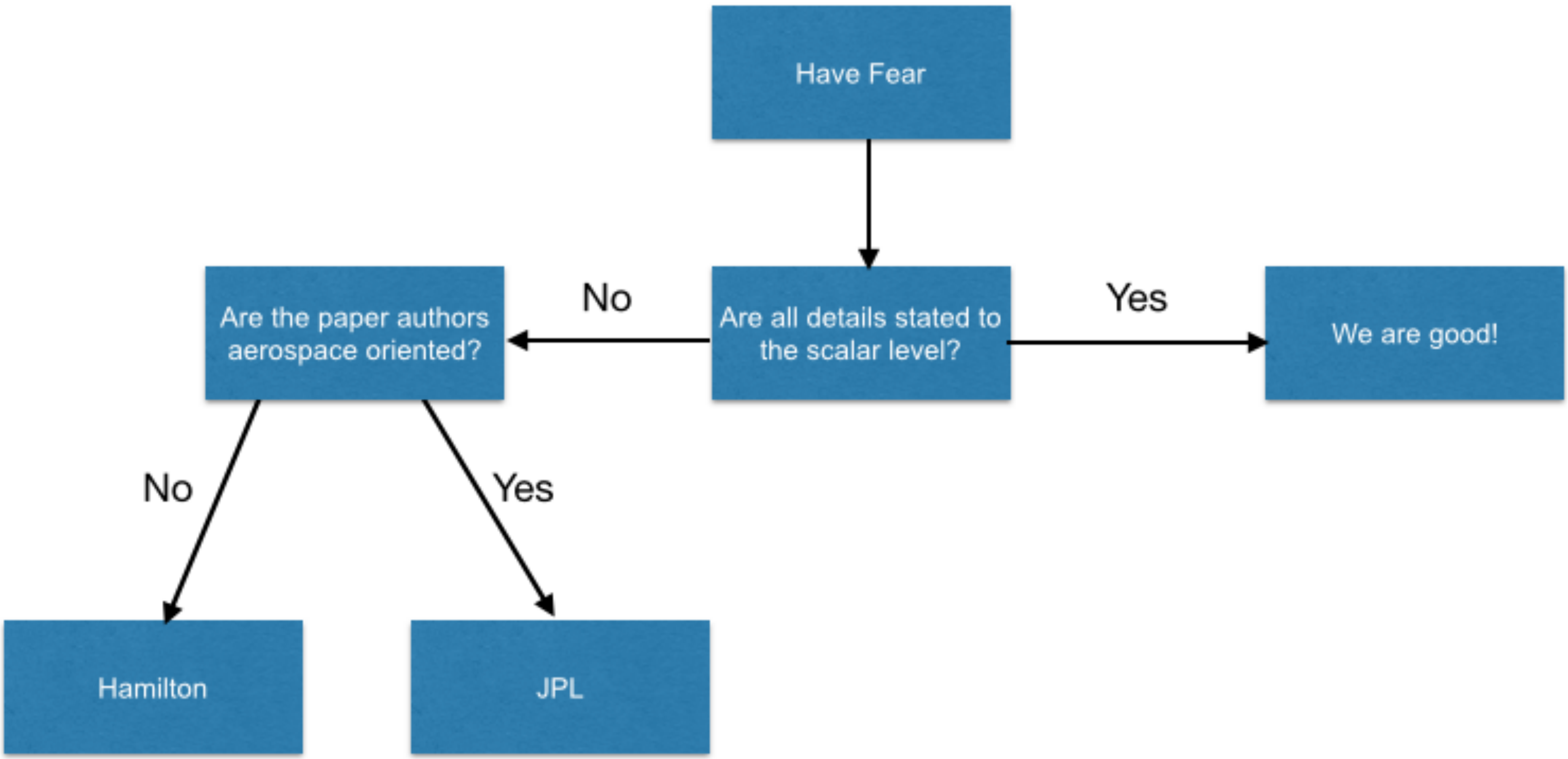
Quaternion decision tree



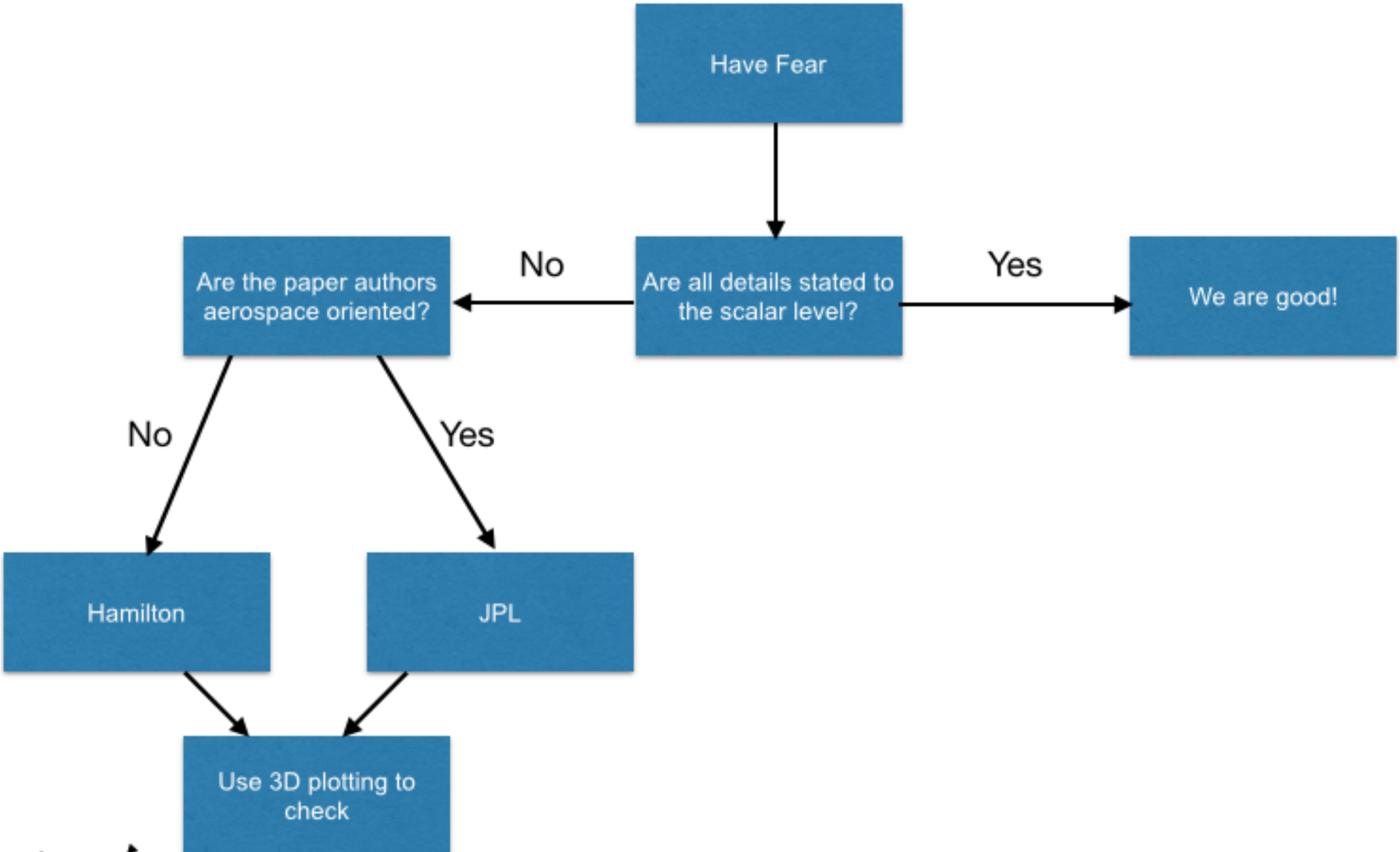
Quaternion decision tree



Quaternion decision tree



Quaternion decision tree



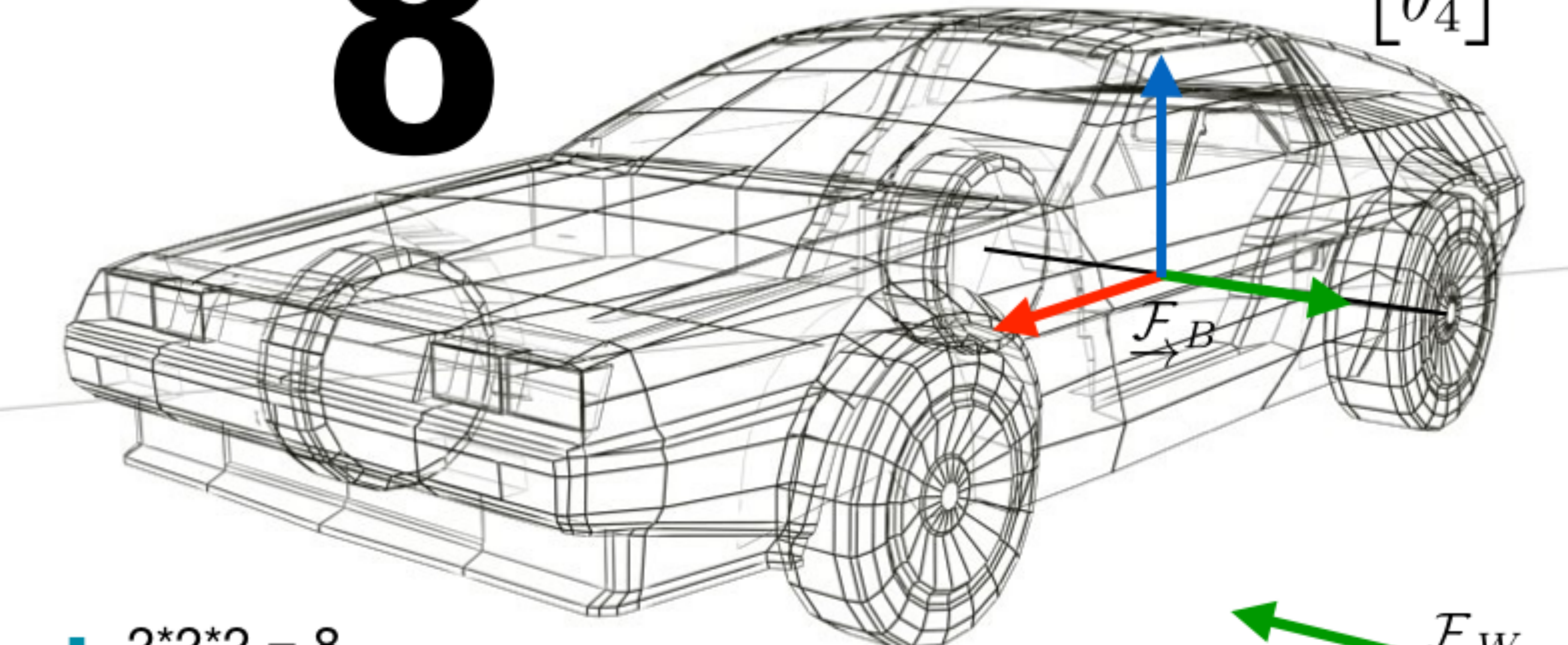
The Bad: Quaternions

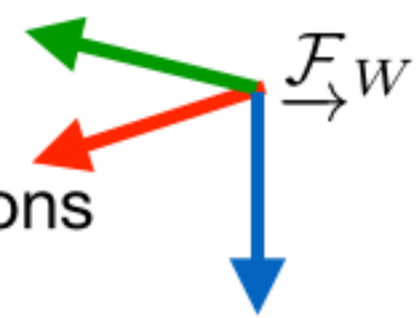
- Unit-length quaternions
  - (2) scalar position
  - (2) body-to-world or world-to-body
  - (2) Hamilton or JPL

- The “four number” problem
- How many possibilities are there?

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix}$$

8





- 2*2*2 = 8
- Restrict the discussion to unit-length quaternions

Recommendation 3

Be clear about your orientation representation.

Recommendation 3

Be clear about your orientation representation.

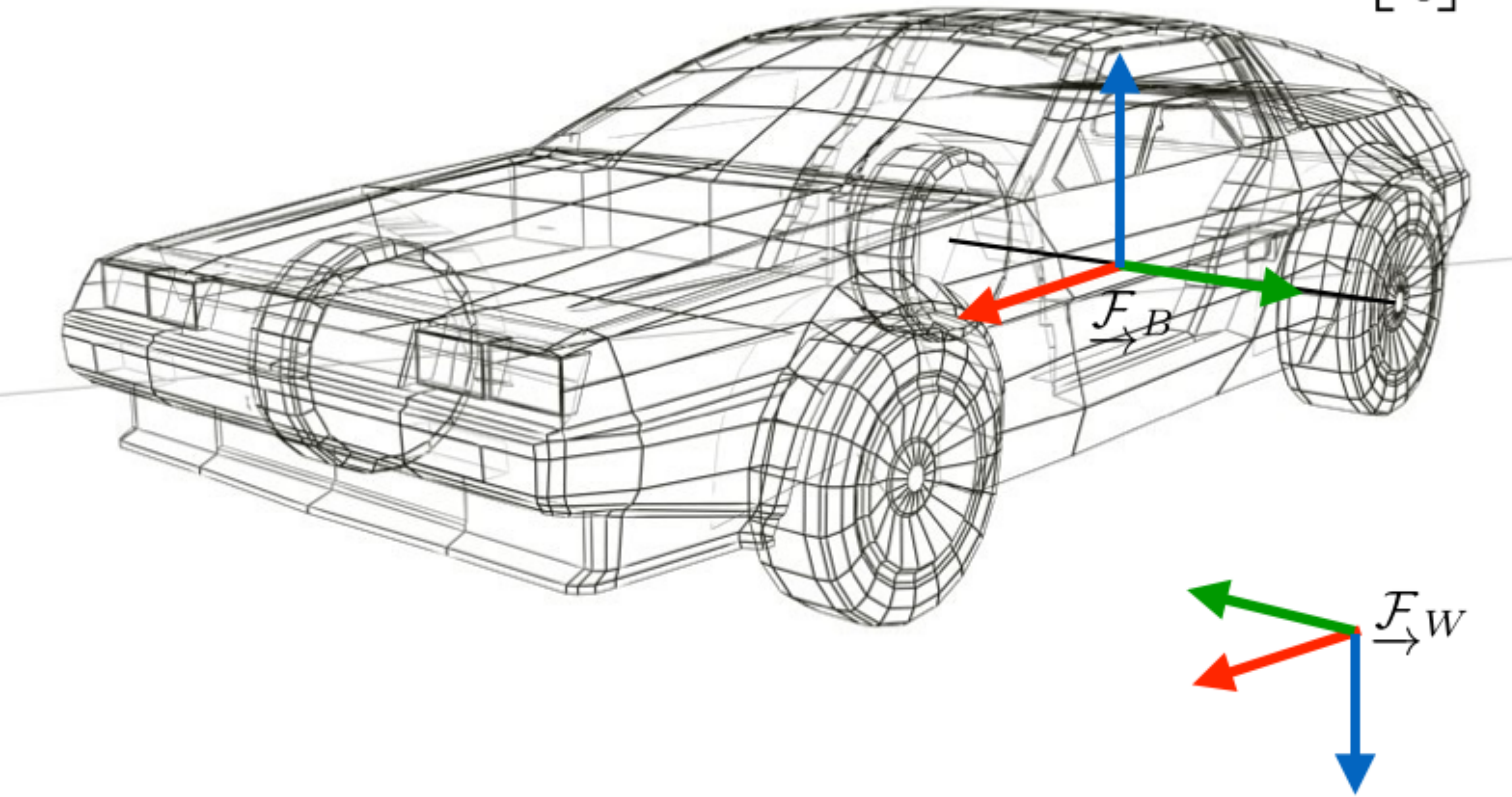
1. **How to build a rotation matrix.**
2. **What rotation matrix that is.**

But translations are easier, right?

Yes and no.

- Another "three number" problem
- How many possibilities are there?

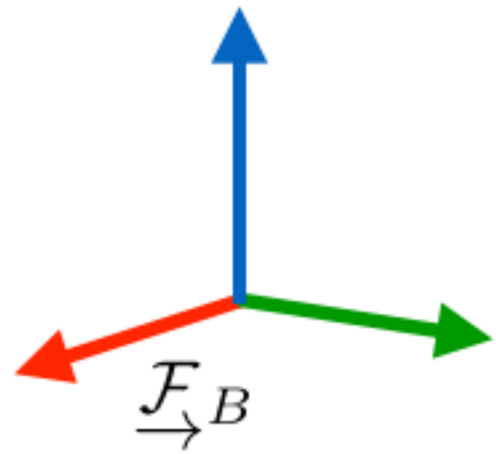
$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$



The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$$_{\mathtt{B}}\mathbf{t}_{\mathtt{BW}}$$

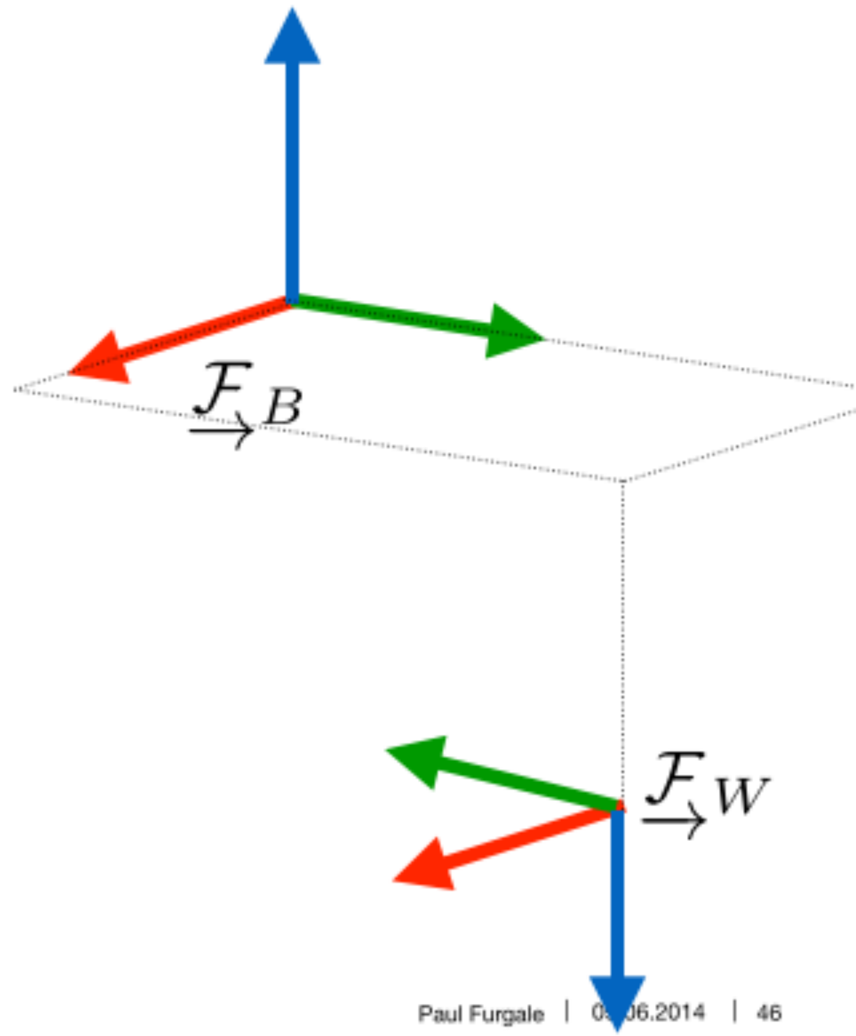


The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$${}_{B}\mathbf{t}_{BW}$$

Expressed in



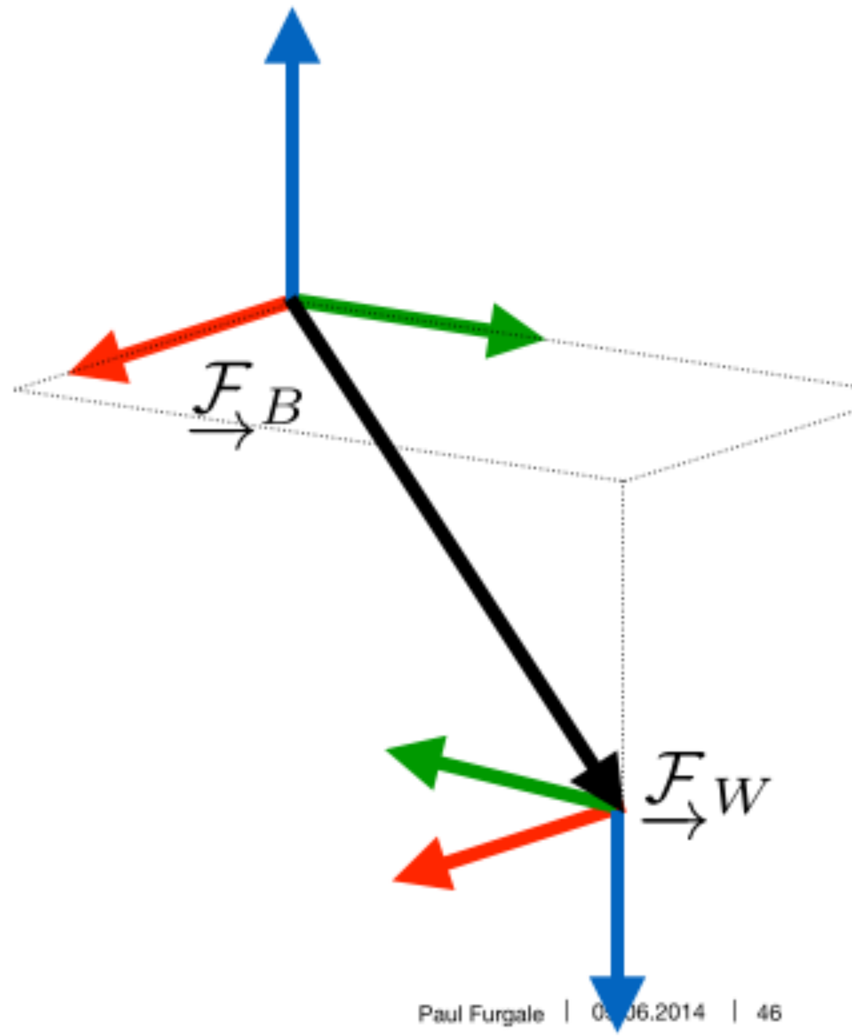
The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$${}_{\mathtt{B}}\mathbf{t}_{\mathtt{BW}}$$

Expressed in

From

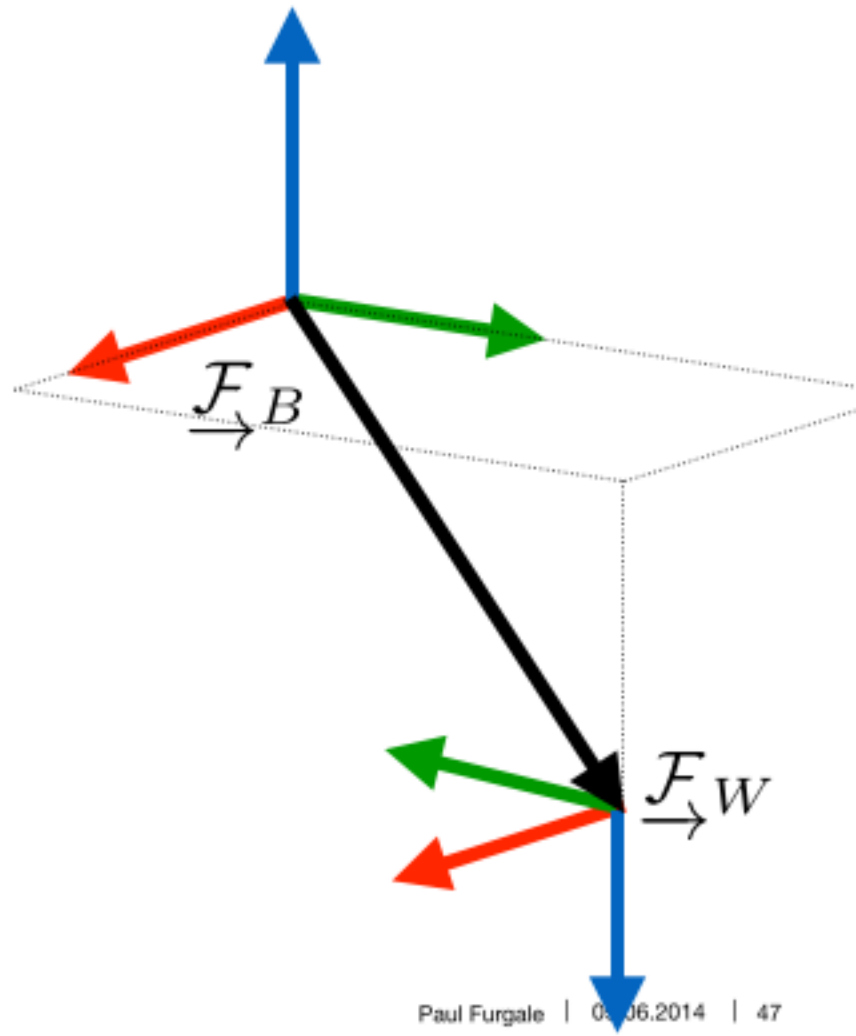


The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$${}_{B}\mathbf{t}_{BW}$$

Expressed in — From — To

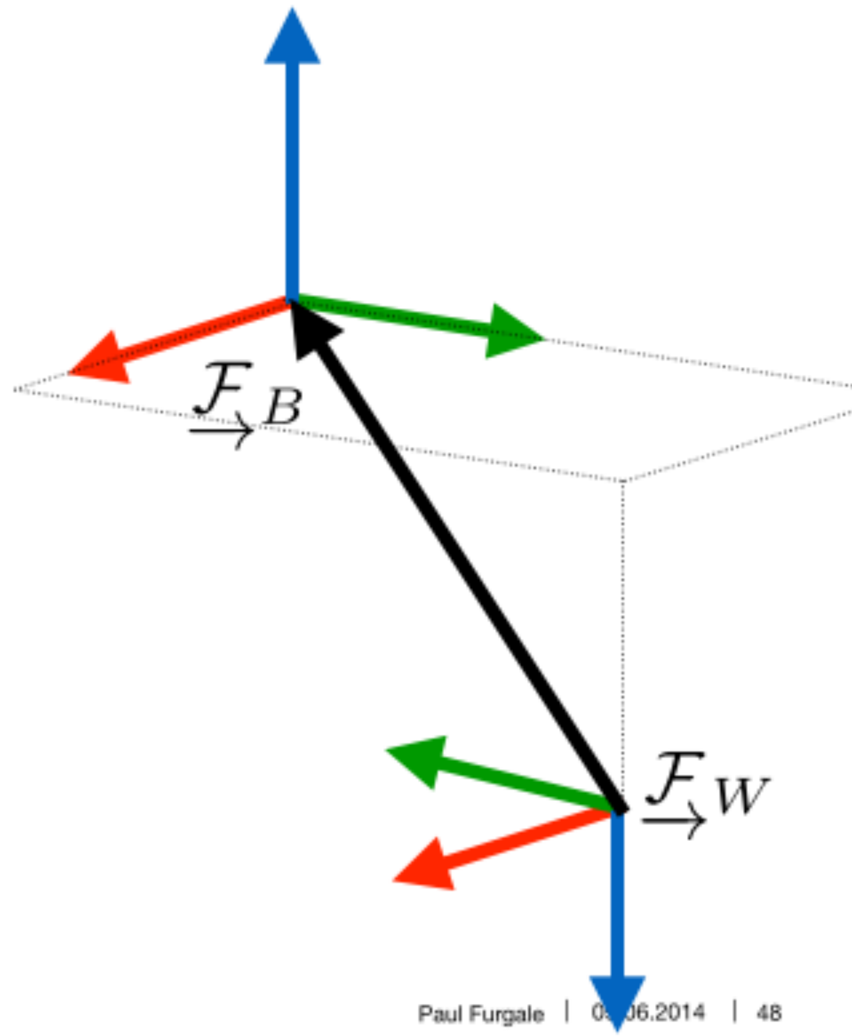


The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$${}_{\mathtt{B}}\mathbf{t}_{\mathtt{WB}}$$

Expressed in — From — To

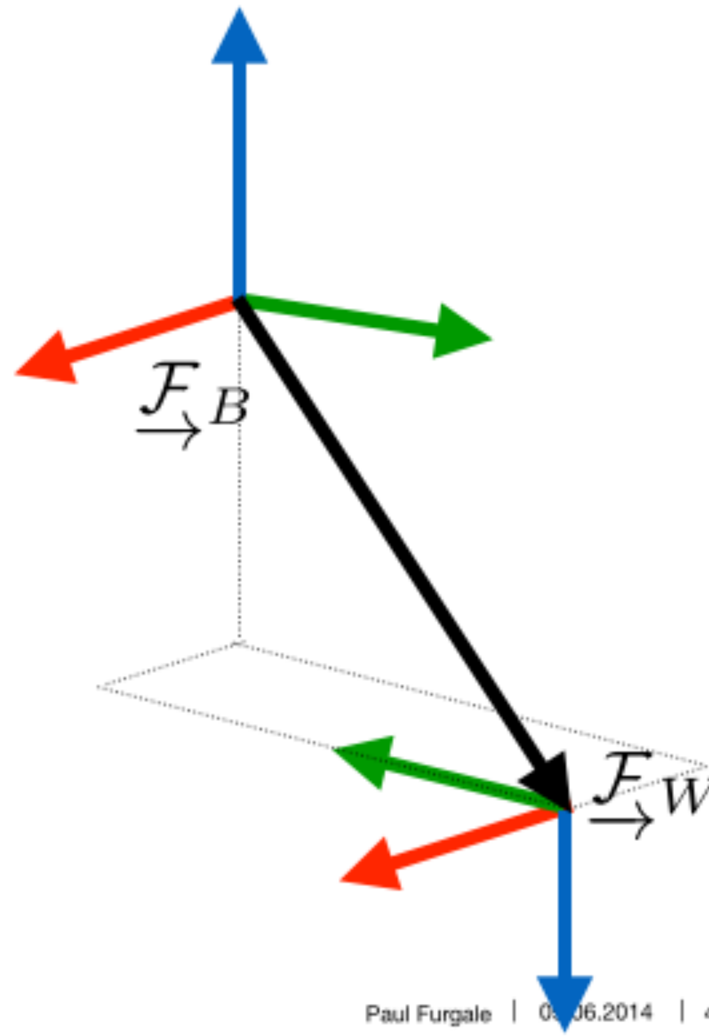


The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$${}_{W}\mathbf{t}_{BW}$$

Expressed in   From   To

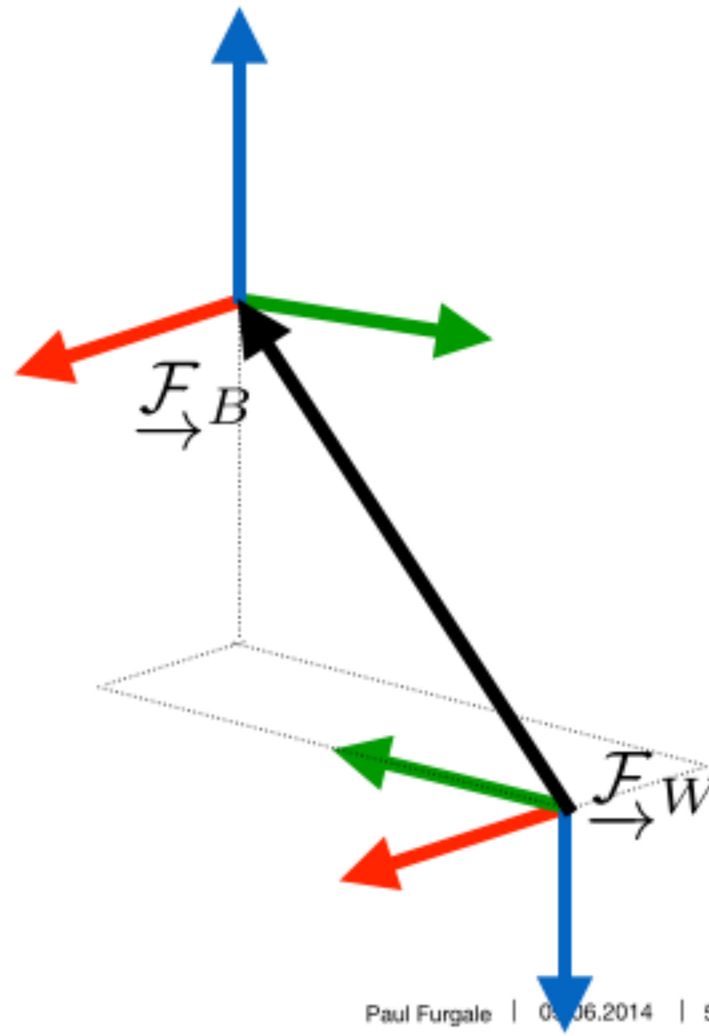


The Bad: translations

$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$${}_{W}\mathbf{t}_{WB}$$

Expressed in — From — To



We will discuss notation soon.

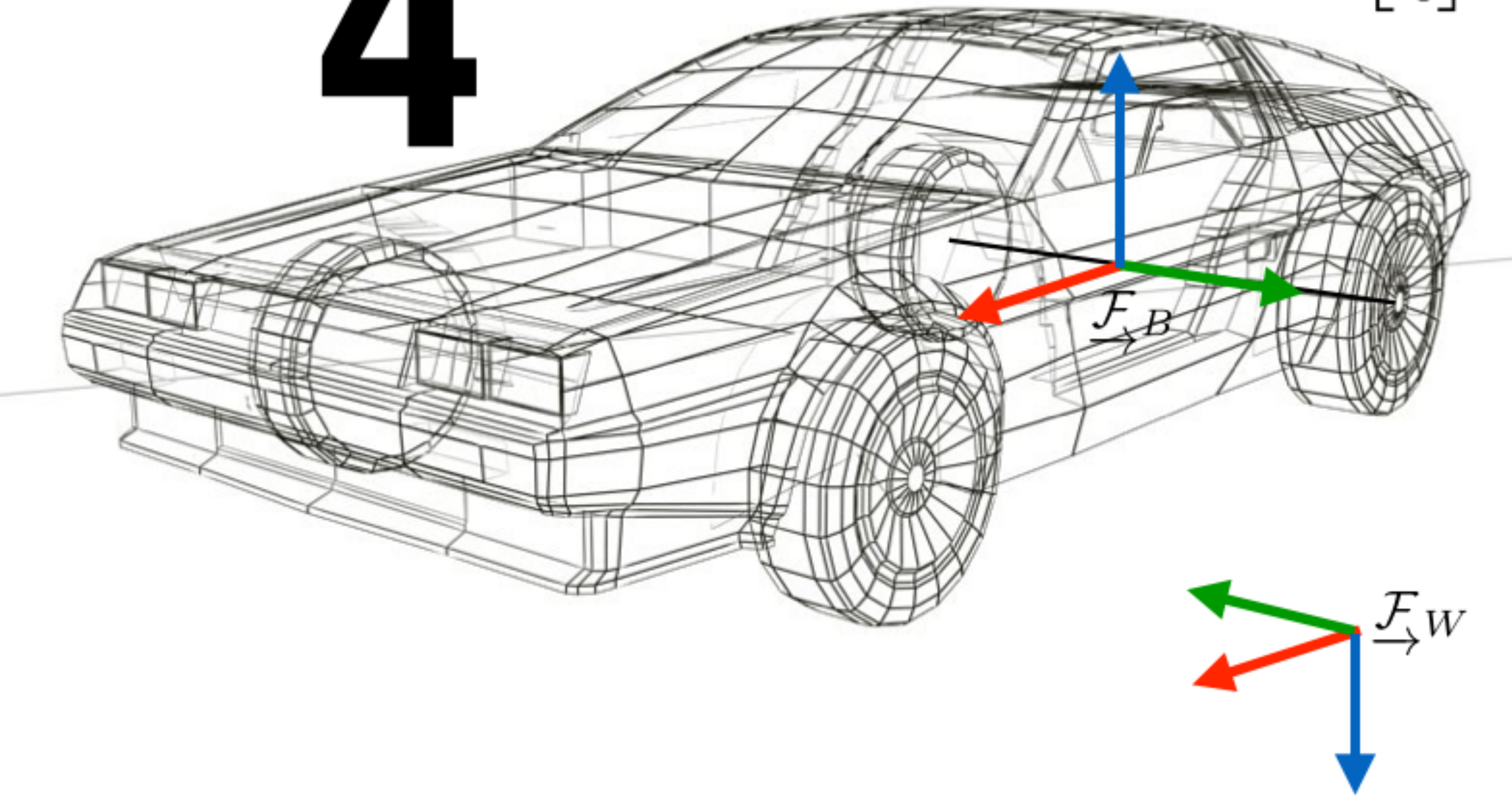
We will discuss notation soon.

What did you think "The Ugly" was?

- Another “three number” problem
- How many possibilities are there?

4

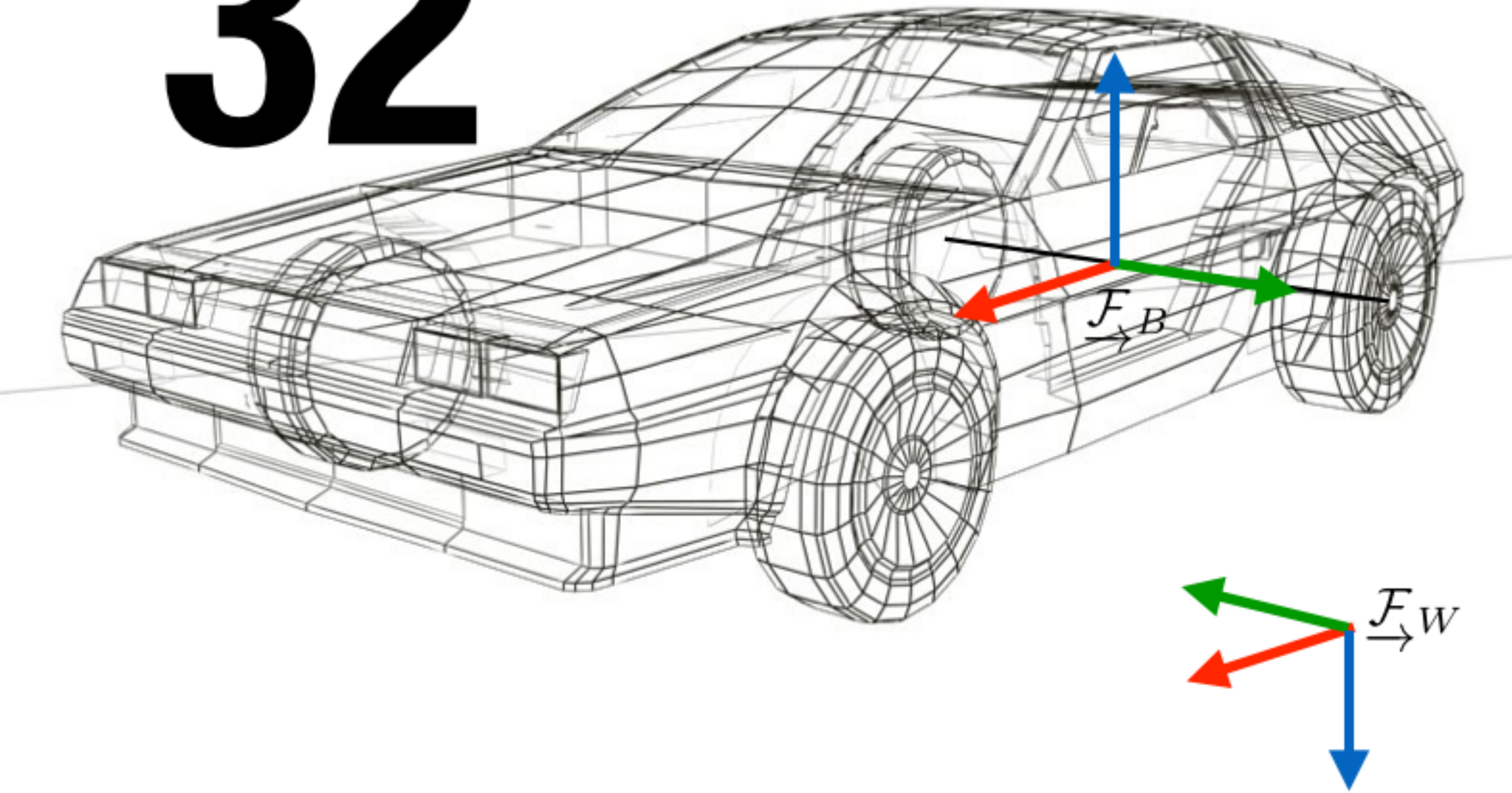
$$\mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$



- With seven numbers for robot pose
- How many possibilities are there?

32

$$\mathbf{C} \leftarrow \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} \quad \mathbf{t} \leftarrow \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$



Recommendation 4

Be clear about your pose representation.

Recommendation 4

Be clear about your pose representation.

1. How to build a transformation matrix.
2. What transformation matrix that is.

Suggested minimum documentation

- Frame diagram.
- Full description of how to build a transformation matrix from the provided scalars and down to the scalar level.
- A clear statement of which transformation matrix it is.

The resulting matrix, $\boldsymbol{T}_{\mathtt{WB}}$, represents the pose of the robot body frame, $\underrightarrow{\mathcal{F}}_{\mathtt{B}}$, with respect to the world frame, $\underrightarrow{\mathcal{F}}_{\mathtt{W}}$, such that a point in the body frame, ${}_{\mathtt{B}}\mathbf{p}$, can be transformed into the world frame by

$$
{}_{\mathtt{W}}\mathbf{p} = \boldsymbol{T}_{\mathtt{WB}}\,{}_{\mathtt{B}}\mathbf{p}. \tag{1}
$$

Experiment: Can I determine what parameterization is used by ROS?

geometry/ CoordinateFrameConventions

Contents

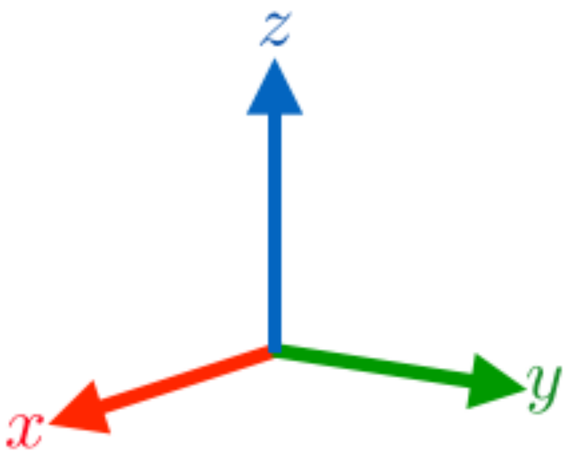
For standardization in ROS we have tried to standardize how information is represented inside a coordinate frame. Below are the conventions which we have chosen:

1. Units

- Distances are in meters
- Angles are in radians

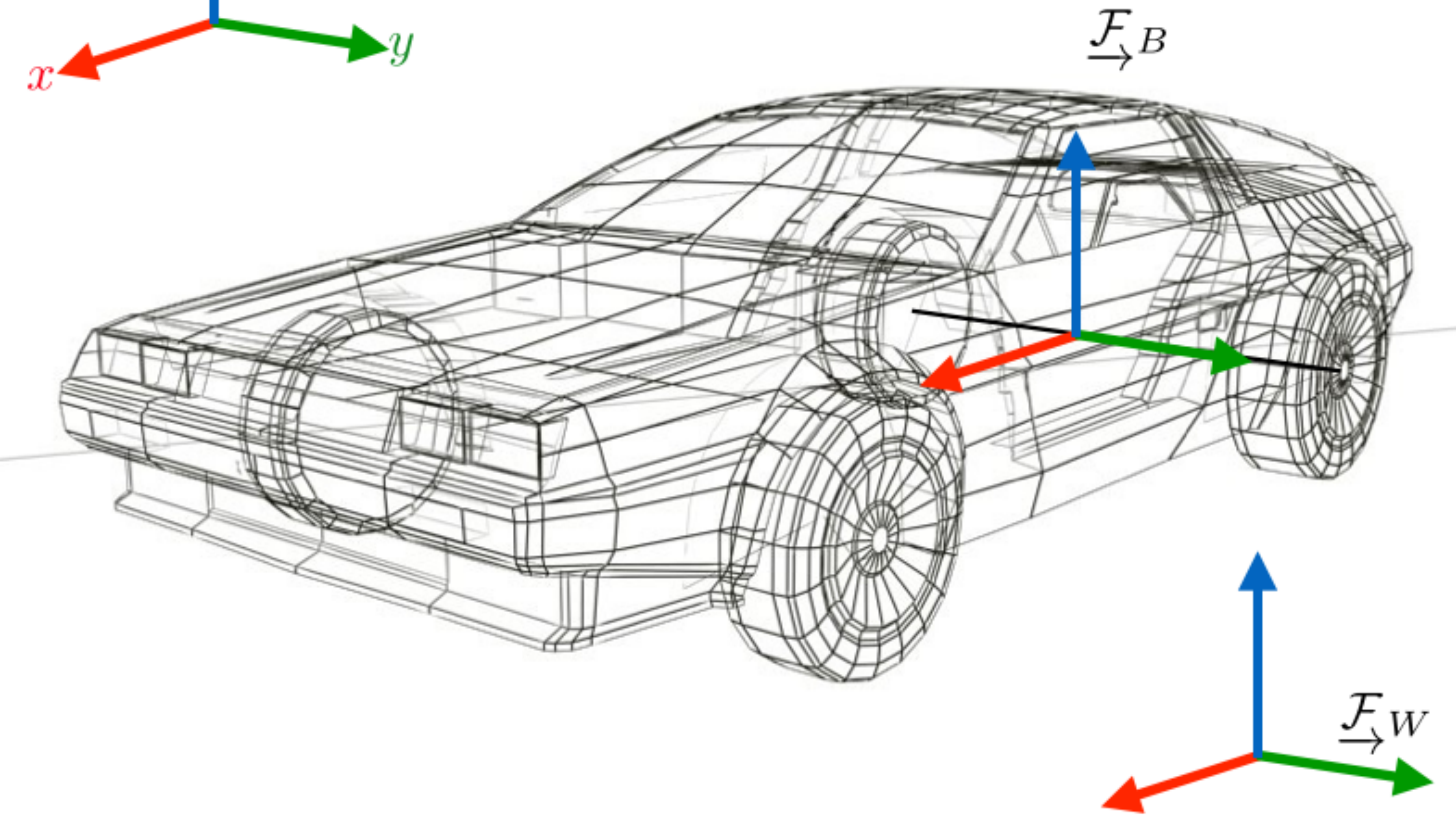
2. Orientation

- X forward
- Y left
- Z up



2. Orientation

- X forward
- Y left
- Z up



geometry_msgs/Quaternion Message

File: geometry_msgs/Quaternion.msg

Raw Message Definition

```
# This represents an orientation in free space in quaternion form.

float64 x
float64 y
float64 z
float64 w
```

Compact Message Definition

```
float64 x
float64 y
float64 z
float64 w
```

autogenerated on Wed, 23 Apr 2014 10:39:28

geometry_msgs/Quaternion Message

File: geometry_msgs/Quaternion.msg

Raw Message Definition

```
# This represents an orientation in free space in quaternion form.

float64 x
float64 y
float64 z
float64 w
```

Compact Message Definition

```
float64 x
float64 y
float64 z
float64 w
```

autogenerated on Wed, 23 Apr 2014 10:39:28

geometry_msgs/Quaternion Message

File: geometry_msgs/Quaternion.msg

Raw Message Definition

```
# This represents an orientation in free space in quaternion form.

float64 x
float64 y
float64 z
float64 w
```

Scalar last

Compact Message Definition

```
float64 x
float64 y
float64 z
float64 w
```

autogenerated on Wed, 23 Apr 2014 10:39:28

geometry_msgs/Pose Message

File: geometry_msgs/Pose.msg

Raw Message Definition

```
# A representation of pose in free space, composed of postion and orientation.
Point position
Quaternion orientation
```

Compact Message Definition

```
geometry_msgs/Point position
geometry_msgs/Quaternion orientation
```

autogenerated on Wed, 23 Apr 2014 10:39:28

geometry_msgs/Transform Message

File: geometry_msgs/Transform.msg

Raw Message Definition

```
# This represents the transform between two coordinate frames in free space.

Vector3 translation
Quaternion rotation
```

Compact Message Definition

```
geometry_msgs/Vector3 translation
geometry_msgs/Quaternion rotation
```

autogenerated on Wed, 23 Apr 2014 10:39:28

geometry_msgs/TransformStamped Message

File: geometry_msgs/TransformStamped.msg

Raw Message Definition

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
#
# This message is mostly used by the
# tf package.
# See its documentation for more information.

Header header
string child_frame_id # the frame id of the child frame
Transform transform
```

Compact Message Definition

```
std_msgs/Header header
string child_frame_id
geometry_msgs/Transform transform
```

autogenerated on Wed, 23 Apr 2014 10:39:28

geometry_msgs/TransformStamped Message

File: geometry_msgs/TransformStamped.msg

Raw Message Definition

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
#
# This message is mostly used by the
# tf package.
# See its documentation for more information.

Header header
string child_frame_id # the frame id of the child frame
Transform transform
```

Compact Message Definition

```
std_msgs/Header header
string child_frame_id
geometry_msgs/Transform transform
```

autogenerated on Wed, 23 Apr 2014 10:39:28

3. Frame Poses

Good stuff on this page!

The relationship between two frames is represented by a 6 DOF relative pose, a translation followed by a rotation. If W and A are two frames, the pose of A in W is given by the translation from W's origin to A's origin, and the rotation of A's coordinate axes in W.

The translation is a vector in W's coordinates, ${}^{W}t_{A}$. It is represented by `tf::Vector3`, which is equivalent to `btVector3`.

The rotation of A is given by a rotation matrix, represented as ${}^{W}_{A}R$, using our convention of the reference frame as a preceeding superscript. The way to read this is: "the rotation of the frame A in W's coordinate system." The columns of R are formed from the three unit vectors of A's axes in W: ${}^{W}X_{A}$, ${}^{W}Y_{A}$, and ${}^{W}Z_{A}$.

There is no `tf` type for a rotation matrix; instead, `tf` represents rotations via `tf::Quaternion`, equivalent to `btQuaternion`. The bullet quaternion type has methods for creating quaternions from rotation matrices, and vice versa.

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame A relative to frame W." The relative pose is constructed as follows:

$$\begin{matrix} {}^{W}_{A}R & {}^{W}t_{A} \\ 0 & 1 \end{matrix}$$

In `tf`, relative poses are represented as `tf::Pose`, which is equivalent to the bullet type `btTransform`. The member functions are `getRotation()` or `getBasis()` for the rotation, and `getOffset()` for the translation of the pose. See the bullet btTransform class reference.

4. Frame poses as Point Mappings

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in A's frame to W." The syntax gives a "cancellation" of the A frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

Mappings or transforms have their own type, `tf::Transform`. This is equivalent to the bullet `btTransform`, so essentially pose offsets and transforms are the same type. Transforms can be created using rotation matrices or quaternions for the rotation, and vectors for the translation. See the bullet btTransform class reference.

3. Frame Poses

Good stuff on this page!

The relationship between two frames is represented by a 6 DOF relative pose, a translation followed by a rotation. If *W* and *A* are two frames, the pose of *A* in *W* is given by the translation from *W*'s origin to *A*'s origin, and the rotation of *A*'s coordinate axes in *W*.

The translation is a vector in *W*'s coordinates, ${}^{W}t_{A}$. It is represented by `tf::Vector3`, which is equivalent to `btVector3`.

The rotation of *A* is given by a rotation matrix, represented as ${}^{W}_{A}R$, using our convention of the reference frame as a preceeding superscript. The way to read this is: "the rotation of the frame *A* in *W*'s coordinate system." The columns of *R* are formed from the three unit vectors of *A*'s axes in *W*: ${}^{W}X_{A}$, ${}^{W}Y_{A}$, and ${}^{W}Z_{A}$.

There is no `tf` type for a rotation matrix; instead, `tf` represents rotations via t Notation n, equivalent to `btQuaternion`. The bullet quaternion type has methods for creating quaterni n matrices and vice versa.

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

$$\begin{matrix} {}^{W}_{A}R & {}^{W}t_{A} \\ 0 & 1 \end{matrix}$$

In `tf`, relative poses are represented as `tf::Pose`, which is equivalent to the bullet type `btTransform`. The member functions are `getRotation()` or `getBasis()` for the rotation, and `getOffset()` for the translation of the pose. See the bullet btTransform class reference.

4. Frame poses as Point Mappings

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

Mappings or transforms have their own type, `tf::Transform`. This is equivalent to the bullet `btTransform`, so essentially pose offsets and transforms are the same type. Transforms can be created using rotation matrices or quaternions for the rotation, and vectors for the translation. See the bullet btTransform class reference.

3. Frame Poses

Good stuff on this page!

The relationship between two frames is represented by a 6 DOF relative pose, a translation followed by a rotation. If W and A are two frames, the pose of A in W is given by the translation from W's origin to A's origin, and the rotation of A's coordinate axes in W.

The translation is a vector in W's coordinates, ${}^{W}t_{A}$. It is represented by `tf::Vector3`, which is equivalent to `btVector3`.

The rotation of A is given by a rotation matrix, represented as ${}^{W}_{A}R$, using our convention of the reference frame as a preceeding superscript. The way to read this is: "the rotation of the frame A in W's coordinate system." The columns of R are formed from the three unit vectors of A's axes in W: ${}^{W}X_{A}$, ${}^{W}Y_{A}$, and ${}^{W}Z_{A}$.

There is no `tf` type for a rotation matrix; instead, `tf` represents rotations via t[illegible]n, equivalent to `btQuaternion`. The bullet quaternion type has methods for creating quaterni[illegible]n matrices and vice versa.

Notation

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame A relative to frame W." The relative pose is constructed as follows:

$$\begin{matrix} {}^{W}_{A}R & {}^{W}t_{A} \\ 0 & 1 \end{matrix}$$

Suggested phrases

In `tf`, relative poses are represented as `tf::Pose`, which is equivalent to the bullet type `btTransform`. The member functions are `getRotation()` or `getBasis()` for the rotation, and `getOffset()` for the translation of the pose. See the bullet btTransform class reference.

4. Frame poses as Point Mappings

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in A's frame to W." The syntax gives a "cancellation" of the A frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

Mappings or transforms have their own type, `tf::Transform`. This is equivalent to the bullet `btTransform`, so essentially pose offsets and transforms are the same type. Transforms can be created using rotation matrices or quaternions for the rotation, and vectors for the translation. See the bullet btTransform class reference.

3. Frame Poses

Good stuff on this page!

The relationship between two frames is represented by a 6 DOF relative pose, a translation followed by a rotation. If *W* and *A* are two frames, the pose of *A* in *W* is given by the translation from *W*'s origin to *A*'s origin, and the rotation of *A*'s coordinate axes in *W*.

The translation is a vector in *W*'s coordinates, ${}^{W}t_{A}$. It is represented by `tf::Vector3`, which is equivalent to `btVector3`.

The rotation of *A* is given by a rotation matrix, represented as ${}^{W}_{A}R$, using our convention of the reference frame as a preceeding superscript. The way to read this is: "the rotation of the frame *A* in *W*'s coordinate system." The columns of *R* are formed from the three unit vectors of *A*'s axes in *W*: ${}^{W}X_{A}$, ${}^{W}Y_{A}$, and ${}^{W}Z_{A}$.

There is no `tf` type for a rotation matrix; instead, `tf` represents rotations via t… Notation …n, equivalent to `btQuaternion`. The bullet quaternion type has methods for creating quaterni… …n matrices and vice versa.

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

Suggested phrases

$$\begin{matrix} {}^{W}_{A}R & {}^{W}t_{A} \\ 0 & 1 \end{matrix}$$

In `tf`, re…e poses are represented as `tf::Pose`, which is equivalent to the bullet type `btTransform`. The member functions are `getRot`… How to build a transformation matrix … the translation of the pose. See the bullet btTransform c…

4. Frame poses as Point Mappings

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

Mappings or transforms have their own type, `tf::Transform`. This is equivalent to the bullet `btTransform`, so essentially pose offsets and transforms are the same type. Transforms can be created using rotation matrices or quaternions for the rotation, and vectors for the translation. See the bullet btTransform class reference.

3. Frame Poses

Good stuff on this page!

The relationship between two frames is represented by a 6 DOF relative pose, a translation followed by a rotation. If W and A are two frames, the pose of A in W is given by the translation from W's origin to A's origin, and the rotation of A's coordinate axes in W.

The translation is a vector in W's coordinates, ${}^{W}t_{A}$. It is represented by `tf::Vector3`, which is equivalent to `btVector3`.

The rotation of A is given by a rotation matrix, represented as ${}^{W}_{A}R$, using our convention of the reference frame as a preceeding superscript. The way to read this is: "the rotation of the frame A in W's coordinate system." The columns of R are formed from the three unit vectors of A's axes in W: ${}^{W}X_{A}$, ${}^{W}Y_{A}$, and ${}^{W}Z_{A}$.

There is no `tf` type for a rotation matrix; instead, `tf` represents rotations via t…n, equivalent to `btQuaternion`. The bullet quaternion type has methods for creating quaterni… matrices and vice versa.

Notation

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame A relative to frame W." The relative pose is constructed as follows:

$$\begin{matrix} {}^{W}_{A}R & {}^{W}t_{A} \\ 0 & 1 \end{matrix}$$

Suggested phrases

In `tf`, rel…e poses are represented as `tf::Pose`, which is equivalent to the bullet type `btTransform`. The member functions are `getRot`… the translation of the pose. See the bullet btTransform c…

How to build a transformation matrix

4. Frame poses as Point Mappings

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in A's frame to W." The syntax gives a "cancellation" of the A frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

Mappings or transforms have their own type, `tf::Transform`. This is equivalent to the bullet `btTransform`, so essentially pose offsets and transform… rotation matrices or quaternions for the rotation, and vectors for the translation. See the bullet btTransform class reference.

What transformation matrix it is

which is equivalent to the bullet type `btTransform`. The member
he rotation, and `getOffset()` for the translation of the pose. See the

t Mappings

g points from one frame to another. The pose ${}^{W}_{A}T$ can also be read
x gives a "cancellation" of the A frame: $\boxed{{}^{W}_{A}T^{A}p = {}^{W}p.}$

`ransform`. This is equivalent to the bullet `btTransform`, so essentially
nsforms can be created using rotation matrices or quaternions for the
let btTransform class reference.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

$$^{W}\mathbf{p} = {}^{W}_{A}\mathbf{T}\,{}^{A}\mathbf{p}$$

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

$$ {}^{W}\mathbf{p} = {}^{W}_{A}\mathbf{T}\,{}^{A}\mathbf{p} $$

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T^{A}p = {}^{W}p$.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

$$^{W}\mathbf{p} = {}^{W}_{A}\mathbf{T}\,^{A}\mathbf{p}$$

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as $\boxed{\text{"the pose of frame } A \text{ relative to frame } W\text{."}}$ The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, $\boxed{\text{"transform a point in } A\text{'s frame to } W\text{."}}$ The syntax gives a "cancellation" of the A frame: ${}^{W}_{A}T\,^{A}p = {}^{W}p$.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

$$ {}^{W}\mathbf{p} = {}^{W}_{A}\mathbf{T}\,{}^{A}\mathbf{p} $$

Frame

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as "the pose of frame A relative to frame W." The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in A's frame to W." The syntax gives a "cancellation" of the A frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

Which interpretation is correct?

Child frame

$$ {}^{W}\mathbf{p} = {}^{W}_{A}\mathbf{T}\,{}^{A}\mathbf{p} $$

Frame

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

Which interpretation is correct?

Frame

$$ {}^{W}\mathbf{p} = {}^{W}_{A}\mathbf{T}\,{}^{A}\mathbf{p} $$

Child frame

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

Recommendation 1

Have fear.

If it is not 100% clear,

assume that you don’t know.

```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

This is the answer.

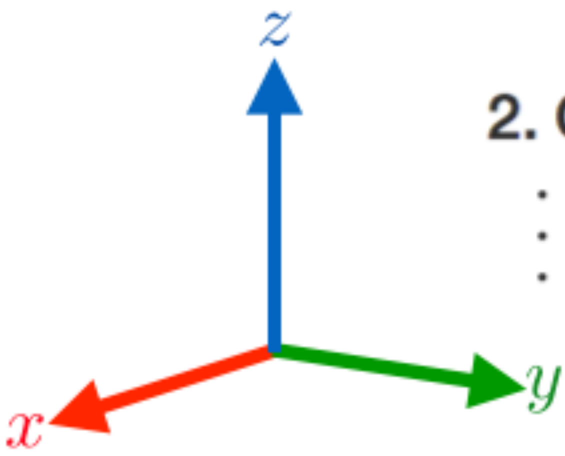
Frame

$$ {}^{B}\mathbf{p} = {}^{B}_{W}\mathbf{T}\,{}^{W}\mathbf{p} $$

Child frame

It's convenient to describe the translation + rotation in homogeneous coordinates, as a single 4x4 matrix ${}^{W}_{A}T$. This can be read as: "the pose of frame *A* relative to frame *W*." The relative pose is constructed as follows:

There is a duality between frame poses and mapping points from one frame to another. The pose ${}^{W}_{A}T$ can also be read as, "transform a point in *A*'s frame to *W*." The syntax gives a "cancellation" of the *A* frame: ${}^{W}_{A}T\,{}^{A}p = {}^{W}p$.

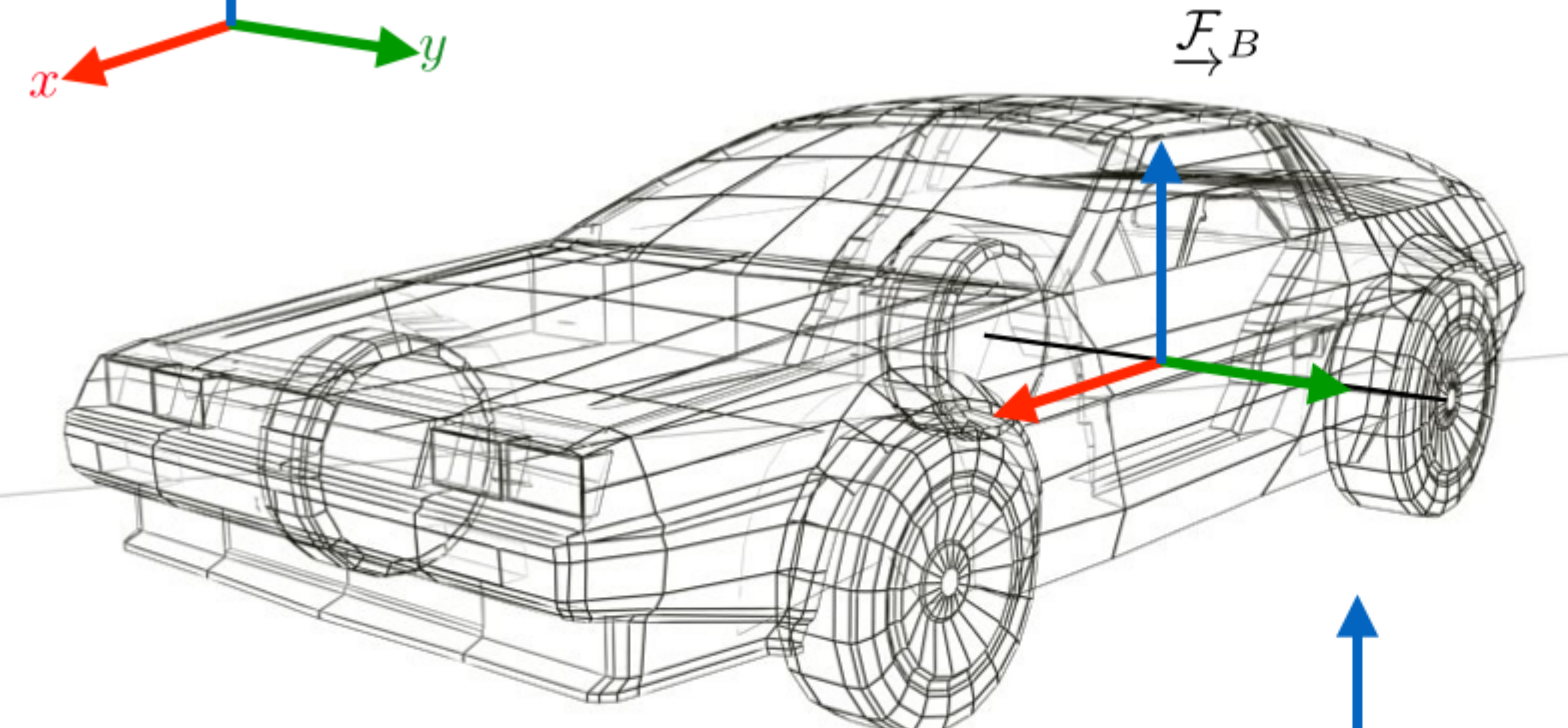


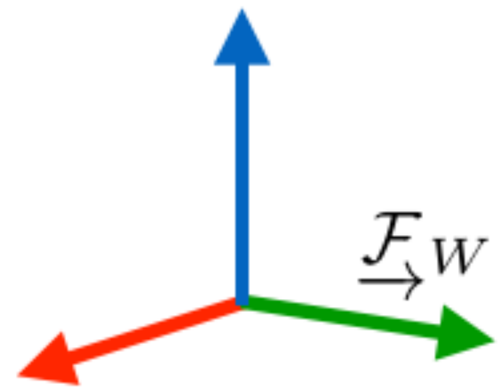
2. Orientation

- X forward
- Y left
- Z up

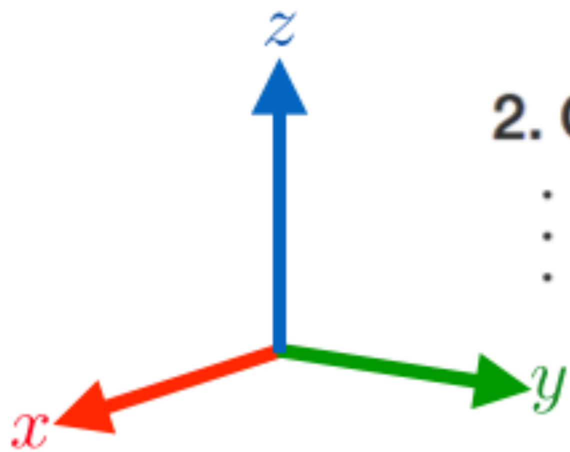
Scalar last

$${}_{B}\mathbf{t}_{BW}, \mathbf{C}_{BW}$$






```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

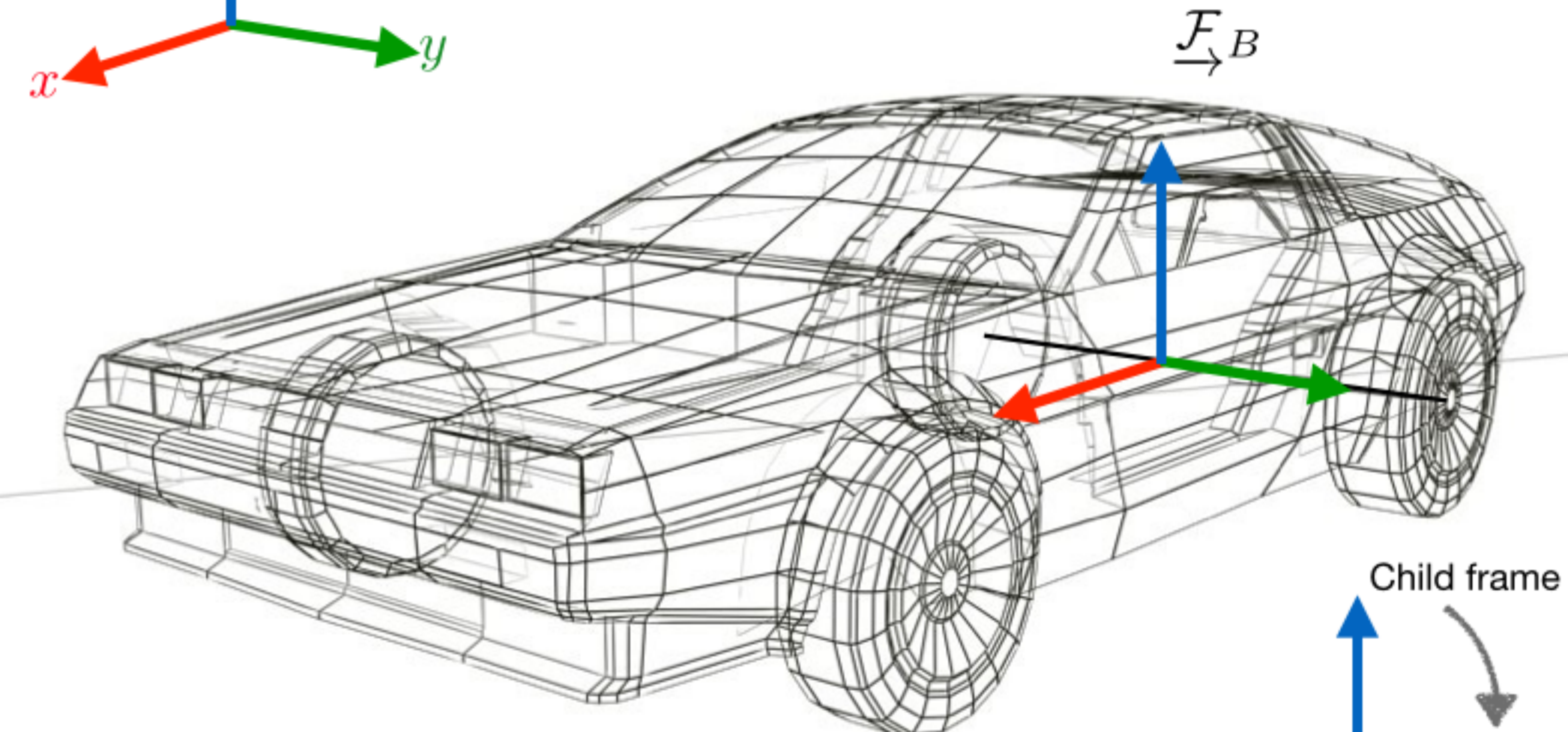


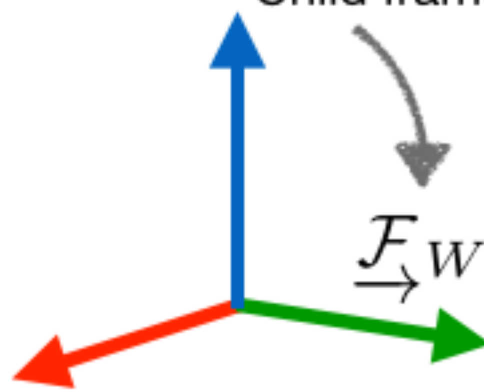
2. Orientation

- X forward
- Y left
- Z up

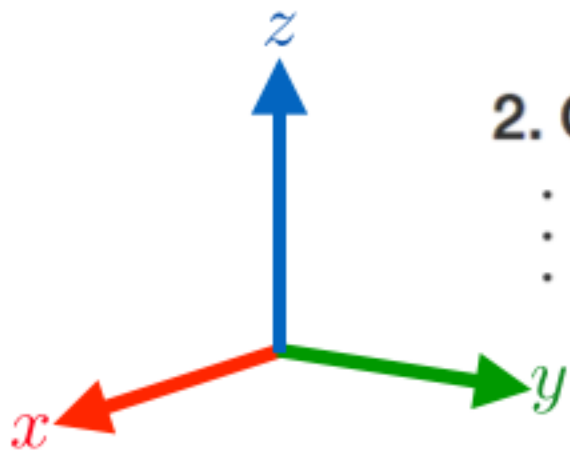
Scalar last

$$ {}_{B}\mathbf{t}_{BW}, \mathbf{C}_{BW} $$






```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

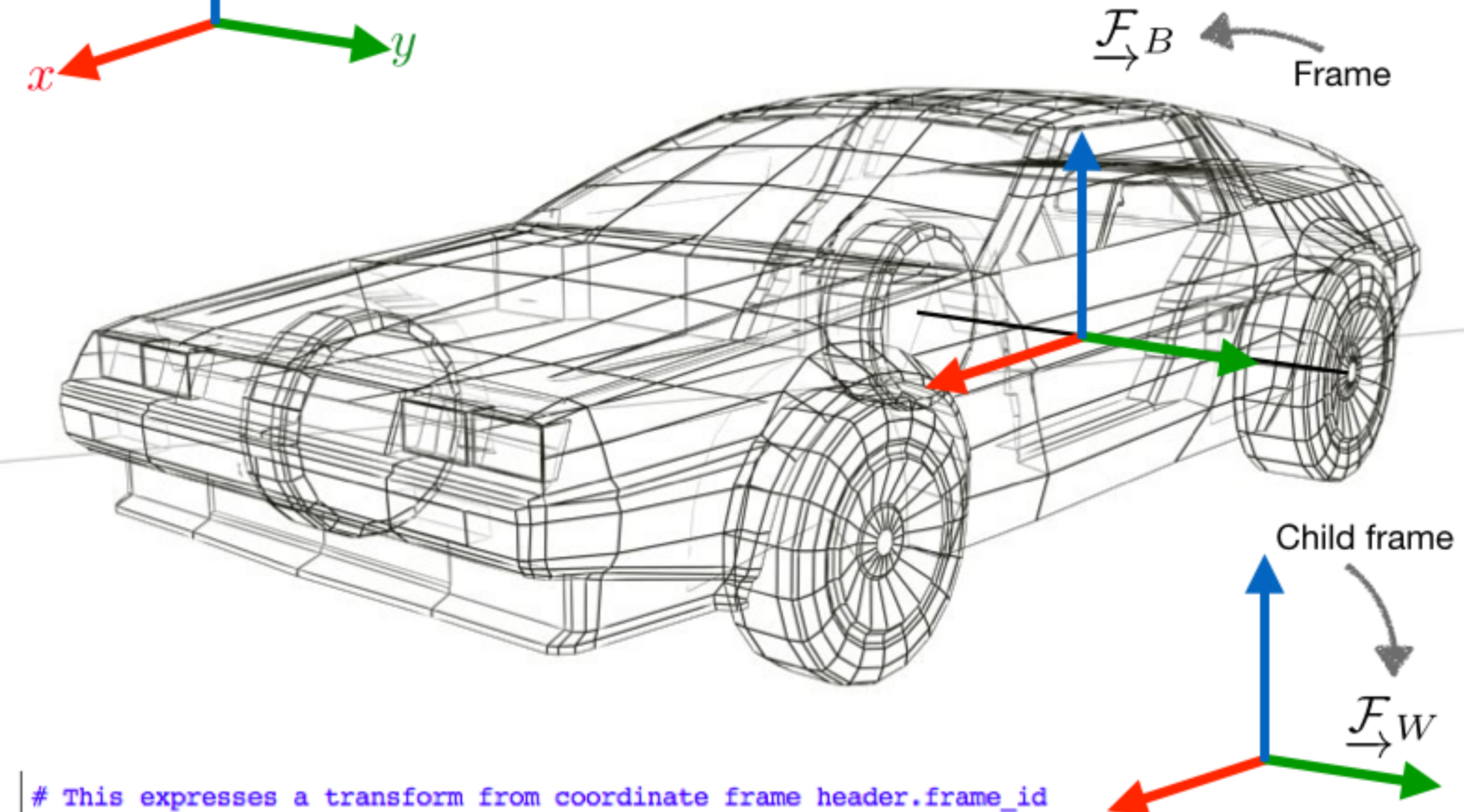


2. Orientation

- X forward
- Y left
- Z up

Scalar last

$${}_{B}\mathbf{t}_{BW},\ \mathbf{C}_{BW}$$




```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

One question remains:
Hamilton or JPL?

geometry_msgs/Quaternion Message

File: geometry_msgs/Quaternion.msg

Raw Message Definition

```
# This represents an orientation in free space in quaternion form.

float64 x
float64 y
float64 z
float64 w
```

Compact Message Definition

```
float64 x
float64 y
float64 z
float64 w
```

autogenerated on Wed, 23 Apr 2014 10:39:28

tf/ Overview/ Data Types

tf API Overview: Data Types | *Transformations and Frames* | *Broadcasting Transforms* | *Using Published Transforms* | *Exceptions*

Contents

1. Data Types

These data types are defined in file `tf/transform_datatypes.h`.

1.1 Base data types (Quaternion, Vector, Point, Pose, Transform)

electric | fuerte | groovy | **hydro** | indigo

As of ROS Fuerte, TF has defined its own datatypes. For more information on how to migrate pre-Fuerte code to newer distributions of ROS, see geometry/bullet_migration.

Type	tf
Quaternion	`tf::Quaternion`
Vector	`tf::Vector3`
Point	`tf::Point`
Pose	`tf::Pose`
Transform	`tf::Transform`

The Quaternion implements quaternion to perform linear algebra rotations in combination with Matrix3x3, Vector3 and Transform.

Definition at line 28 of file Quaternion.h.

Constructor & Destructor Documentation

```
tf::Quaternion::Quaternion ( ) [inline]
```

No initialization constructor.

Definition at line 31 of file Quaternion.h.

```
tf::Quaternion::Quaternion ( const tfScalar &  x,
                             const tfScalar &  y,
                             const tfScalar &  z,
                             const tfScalar &  w
                           )                   [inline]
```

Constructor from scalars.

Definition at line 37 of file Quaternion.h.

tf/ Overview/ Transformations

tf API Overview: Data Types | Transformations and Frames | Broadcasting Transforms | Using Published Transforms | Exceptions

Contents

Coodinate Frames, Transforms, and TF

1. References

An excellent reference for coordinate frames and transforms is the first chapter of John Craig's book, *Introduction to Robotics* [1986, 1989]. We follow his conventions for transform and point indices.

Geometric objects in tf are represented by tf types, which are equivalent to corresponding bullet types; see tf data types. Bullet class references for transforms and quaternions are handy.

tf/ Overview/ Transformations

tf API Overview: Data Types | Transformations and Frames | Broadcasting Transforms | Using Published Transforms | Exceptions

Contents

Coodinate Frames, Transforms, and TF

1. References

An excellent reference for coordinate frames and transforms is the first chapter of John Craig's book, *Introduction to Robotics* [1986, 1989]. We follow his conventions for transform and point indices.

Geometric objects in tf are represented by tf types which are equivalent to corresponding bullet types; see tf data types. Bullet class references for transforms and quaternions are handy.

Detailed Description

The btQuaternion implements quaternion to perform linear algebra rotations in combination with btMatrix3x3, btVector3 and btTransform.

Definition at line 42 of file btQuaternion.h.

Constructor & Destructor Documentation

btQuaternion::btQuaternion ( ) inline

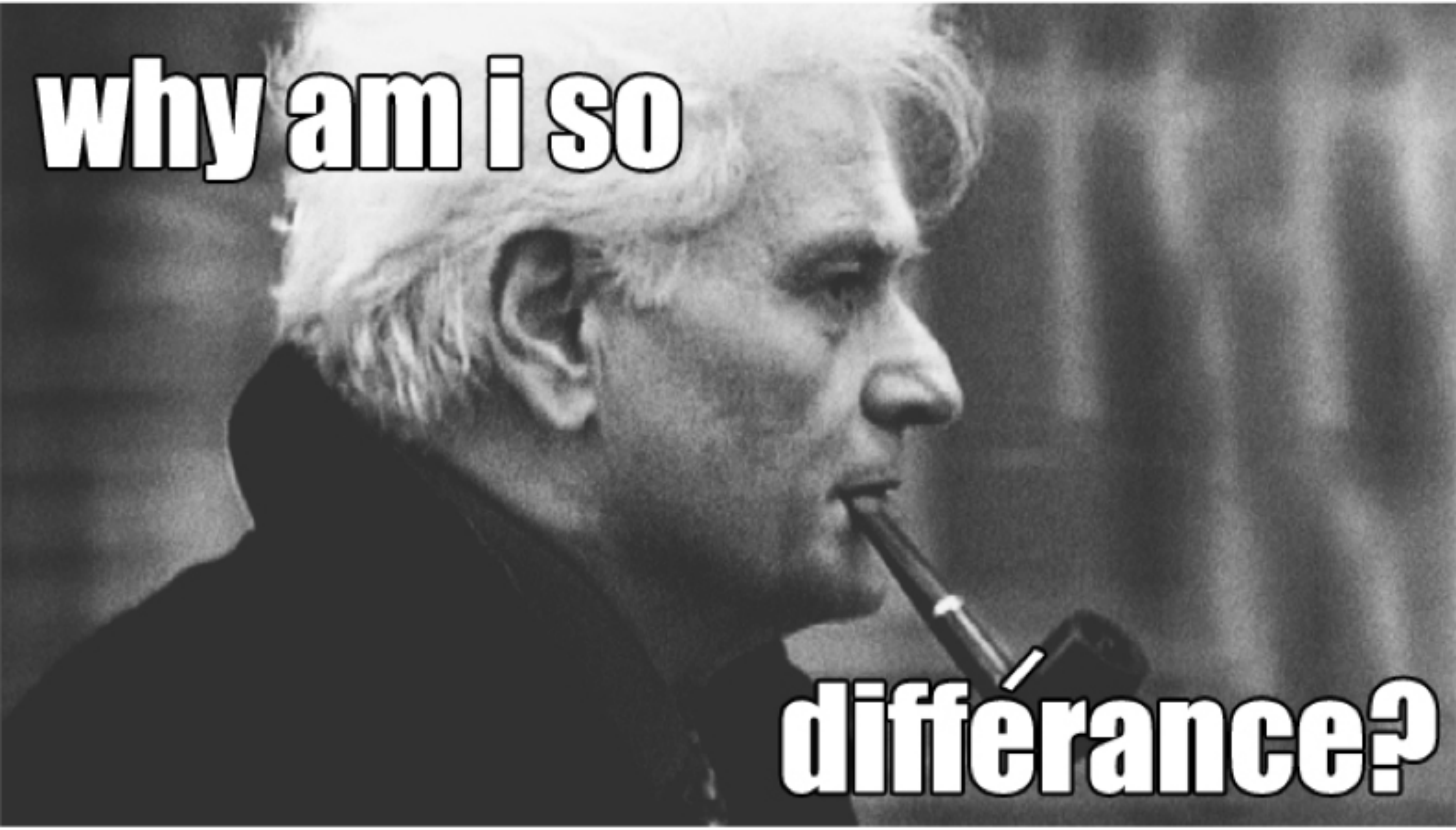
No initialization constructor.

Definition at line 45 of file btQuaternion.h.

```
btQuaternion::btQuaternion ( const btScalar &  _x,
                             const btScalar &  _y,
                             const btScalar &  _z,
                             const btScalar &  _w
                           )
```
inline

Constructor from scalars.

Definition at line 74 of file btQuaternion.h.


why am i so
différance?

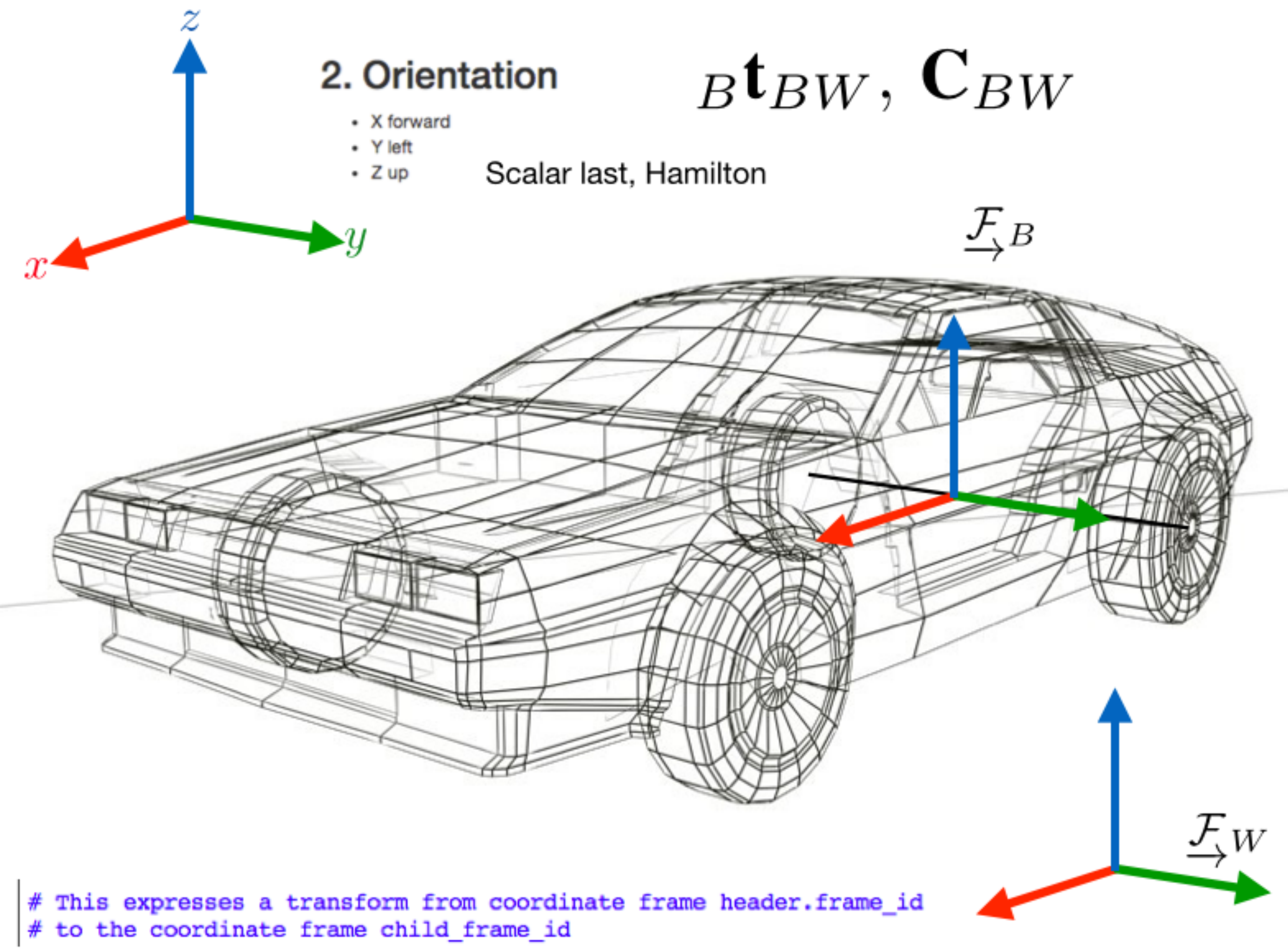
Experiment: Can I determine what parameterization is used by ROS?

No.

Recommendation 4

Be clear about your pose representation.

1. How to build a transformation matrix.
2. What transformation matrix that is.



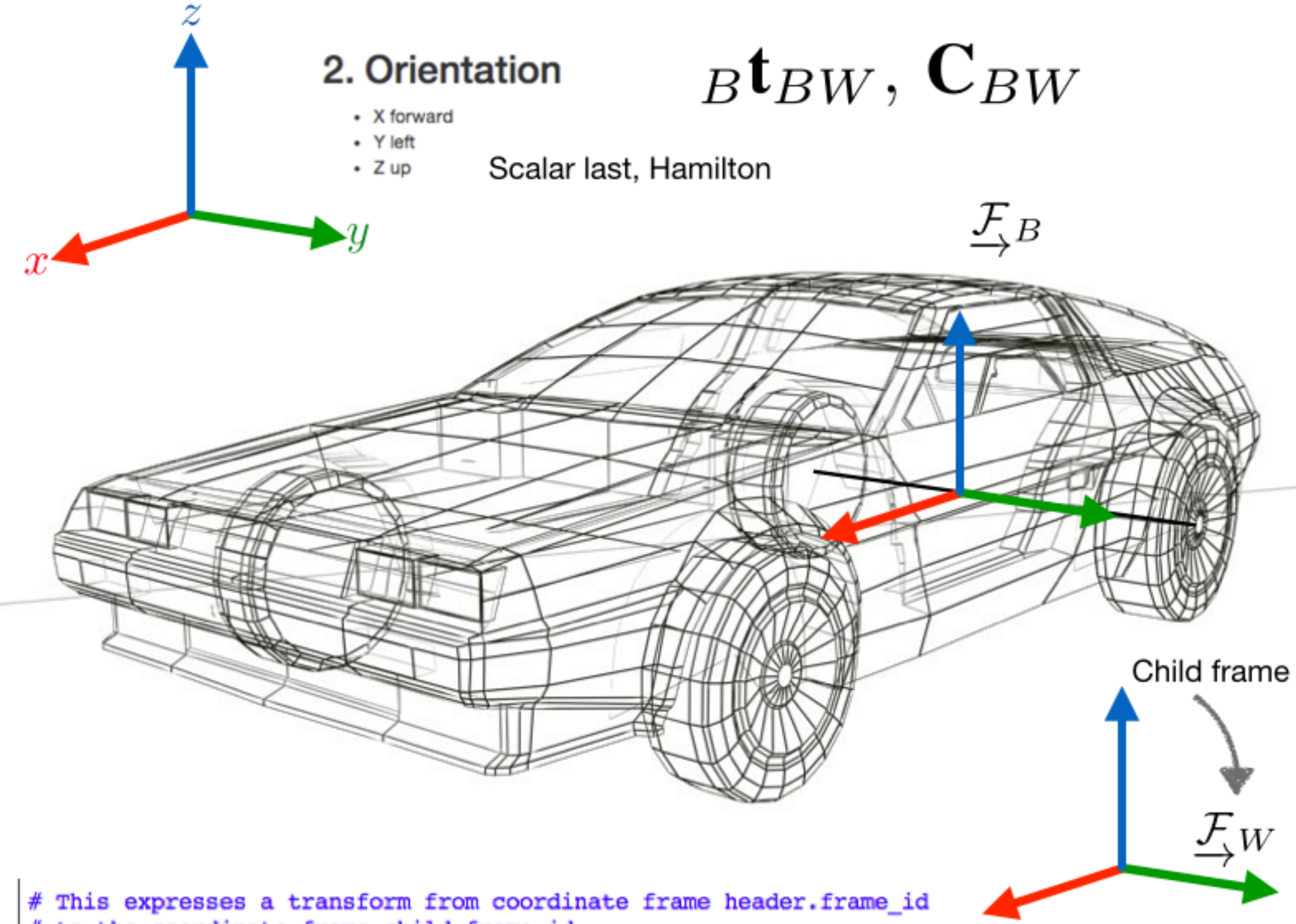
2. Orientation
• X forward
• Y left
• Z up
Scalar last, Hamilton
$_B\mathbf{t}_{BW}, \mathbf{C}_{BW}$
z
y
x
$\underrightarrow{\mathcal{F}}_B$
$\underrightarrow{\mathcal{F}}_W$
This expresses a transform from coordinate frame header.frame_id
to the coordinate frame child_frame_id

2. Orientation

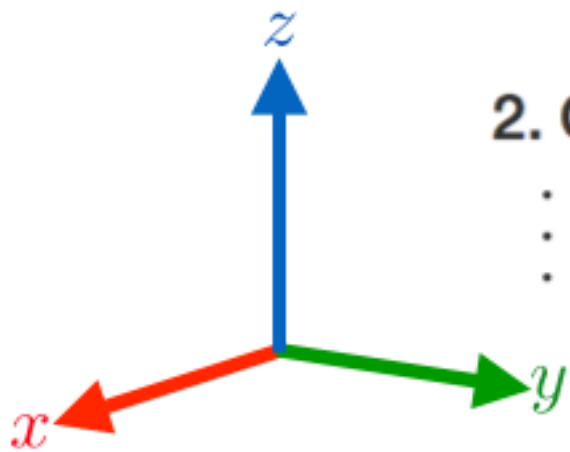
- X forward
- Y left
- Z up

$$_B\mathbf{t}_{BW}, \mathbf{C}_{BW}$$

Scalar last, Hamilton




```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

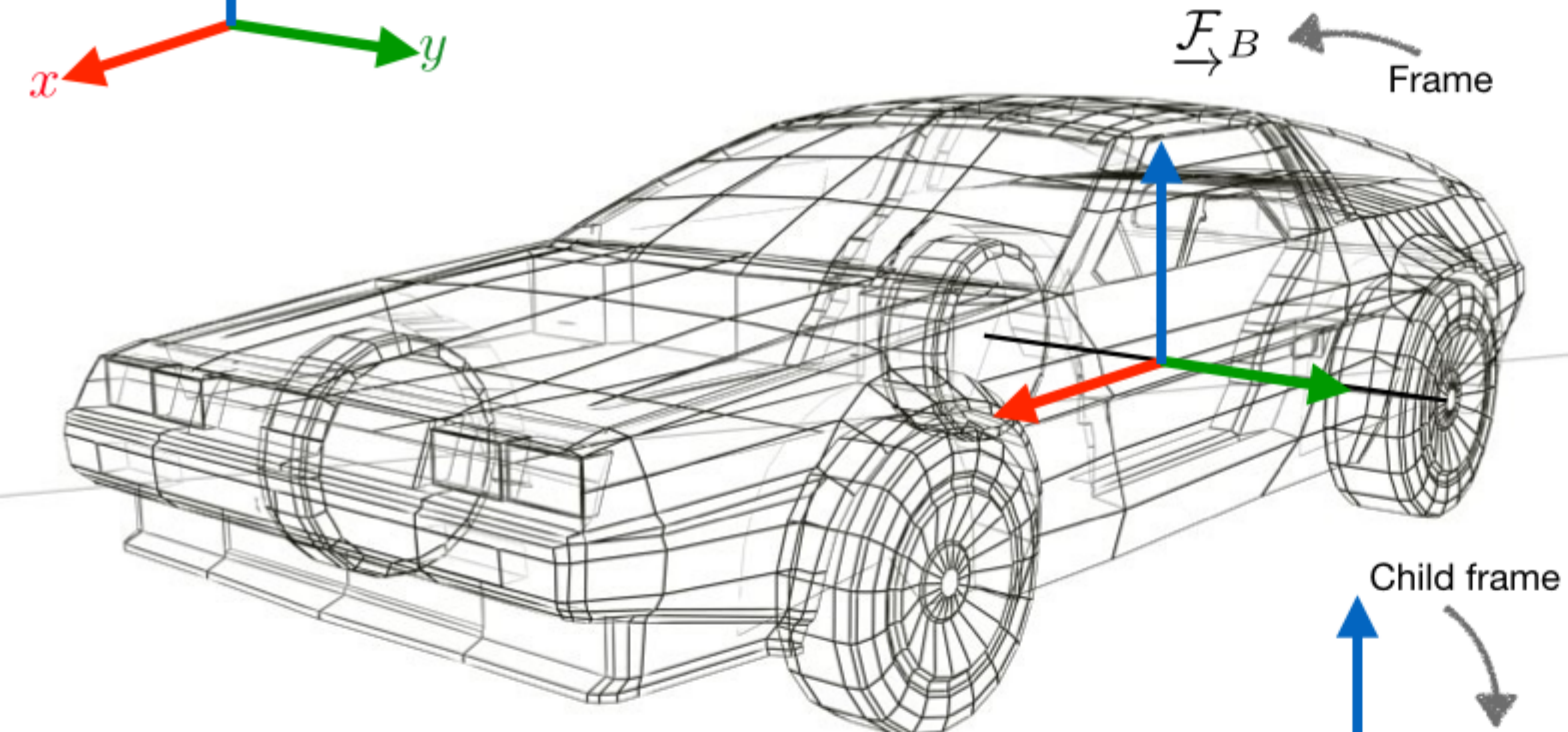


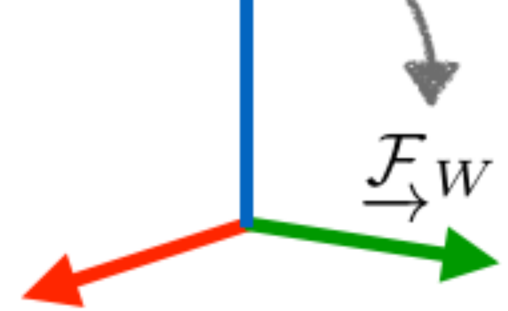
2. Orientation

- X forward
- Y left
- Z up

$$ {}_{B}\mathbf{t}_{BW}\,,\ \mathbf{C}_{BW} $$

Scalar last, Hamilton






```
# This expresses a transform from coordinate frame header.frame_id
# to the coordinate frame child_frame_id
```

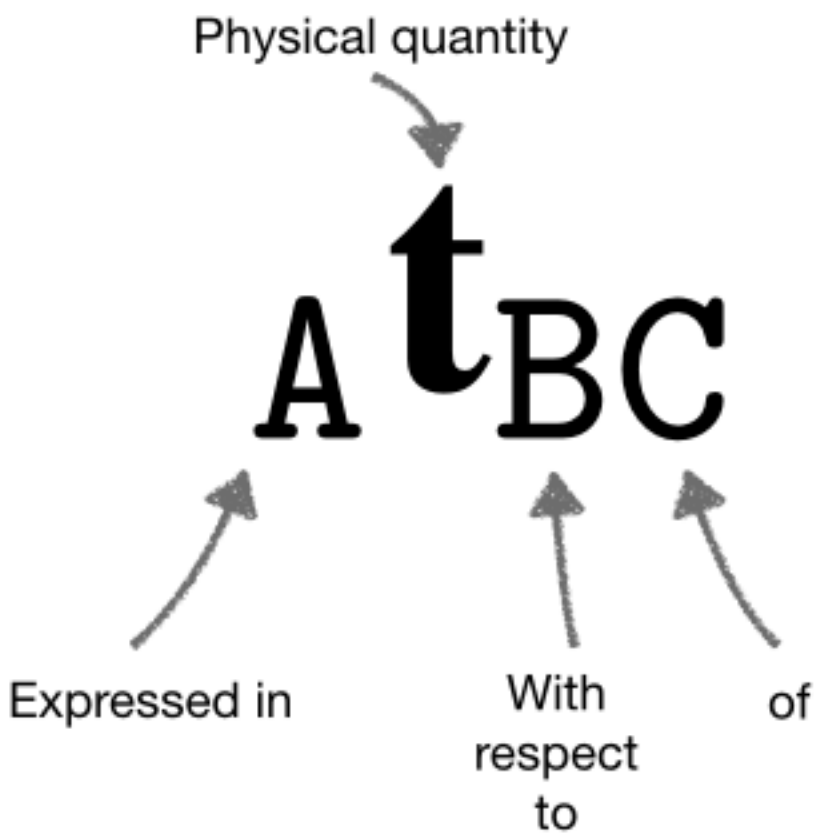
The Ugly

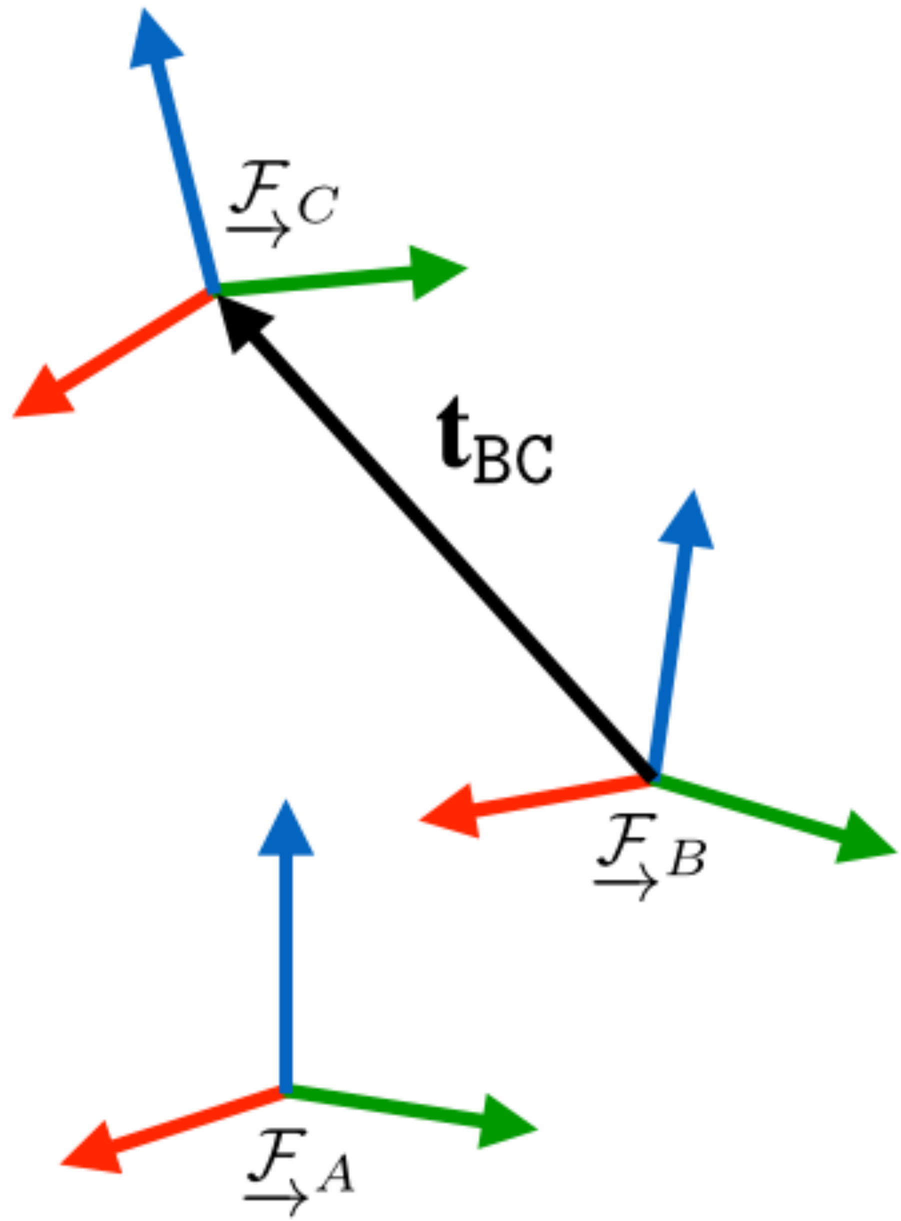
Let’s talk about notation.

The Ugly: Notation

- Minimum required notation to be unambiguous
- Rotation matrices have two frame decorations:
  - to
  - from
- Coordinates of vectors have three decorations:
  - to
  - from
  - expressed in

The Ugly: Notation





The Ugly: Notation

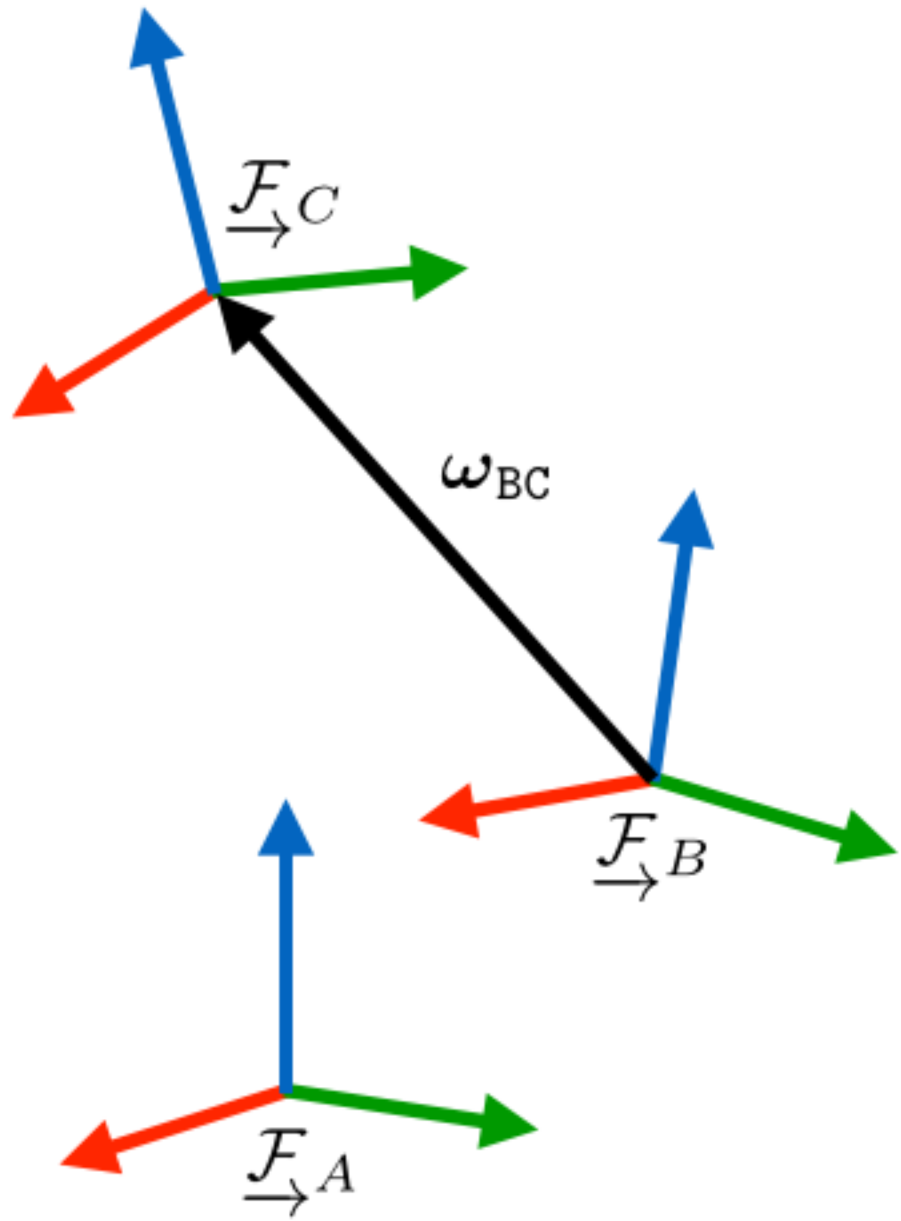
Physical quantity

$${}_{\mathrm{A}}\boldsymbol{\omega}_{\mathrm{BC}}$$

Expressed in

With respect to

of

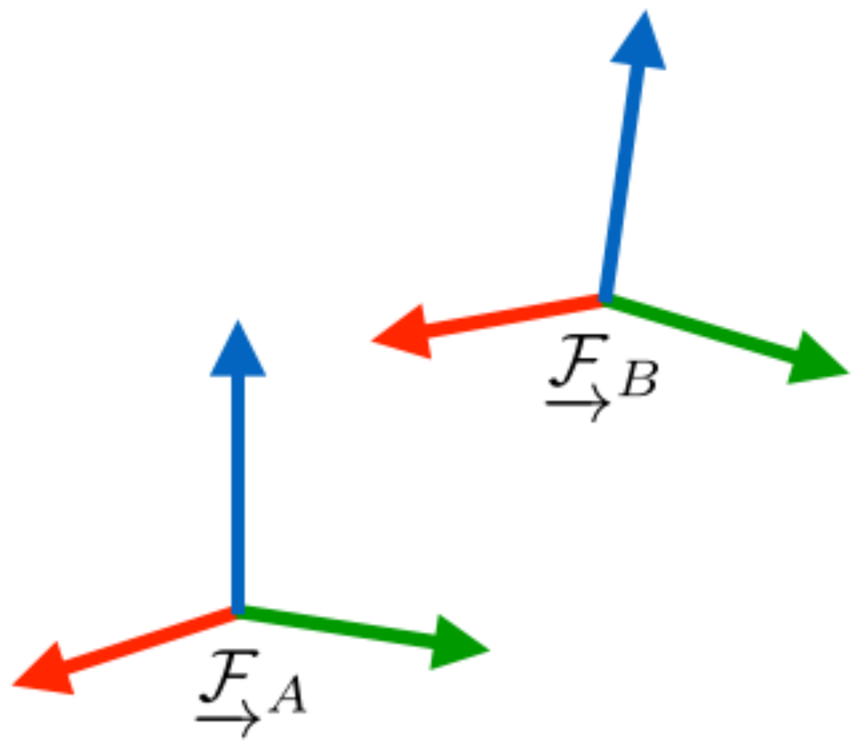


The Ugly: Notation

$$\mathbf{C}_{\mathtt{AB}}$$

into

rotates vectors from



$${}_{\mathtt{A}}\mathbf{v}_{\mathtt{BC}} = \mathbf{C}_{\mathtt{AB}}\,{}_{\mathtt{B}}\mathbf{v}_{\mathtt{BC}}$$

The Ugly: Notation

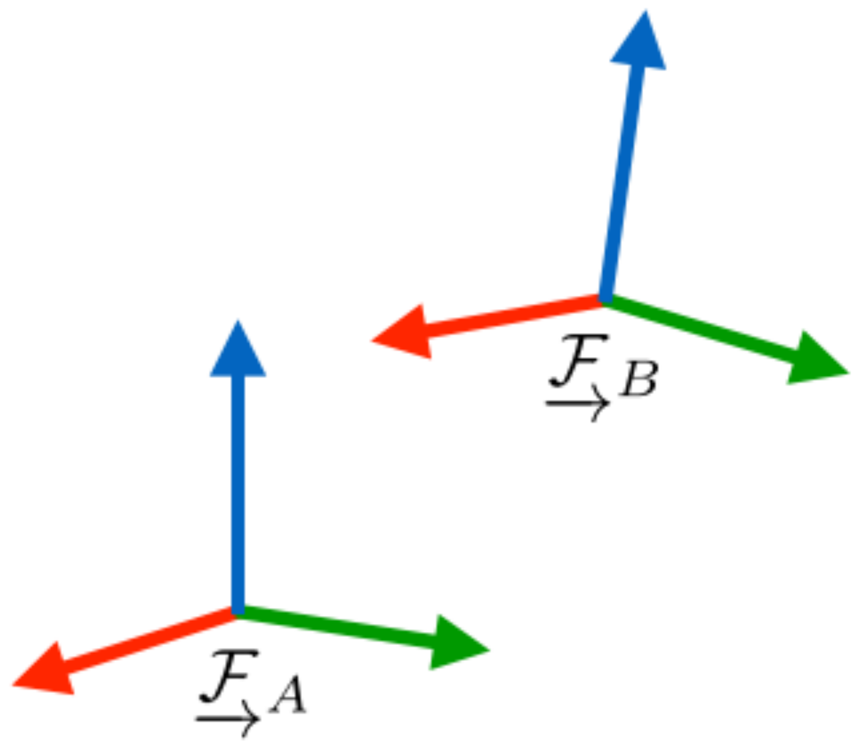
"The orientation/attitude of B with respect to A"

$$\mathbf{C}_{\mathtt{AB}}$$

With respect to

of

$${}_{\mathtt{A}}\mathbf{v}_{\mathtt{BC}} = \mathbf{C}_{\mathtt{AB}}\,{}_{\mathtt{B}}\mathbf{v}_{\mathtt{BC}}$$



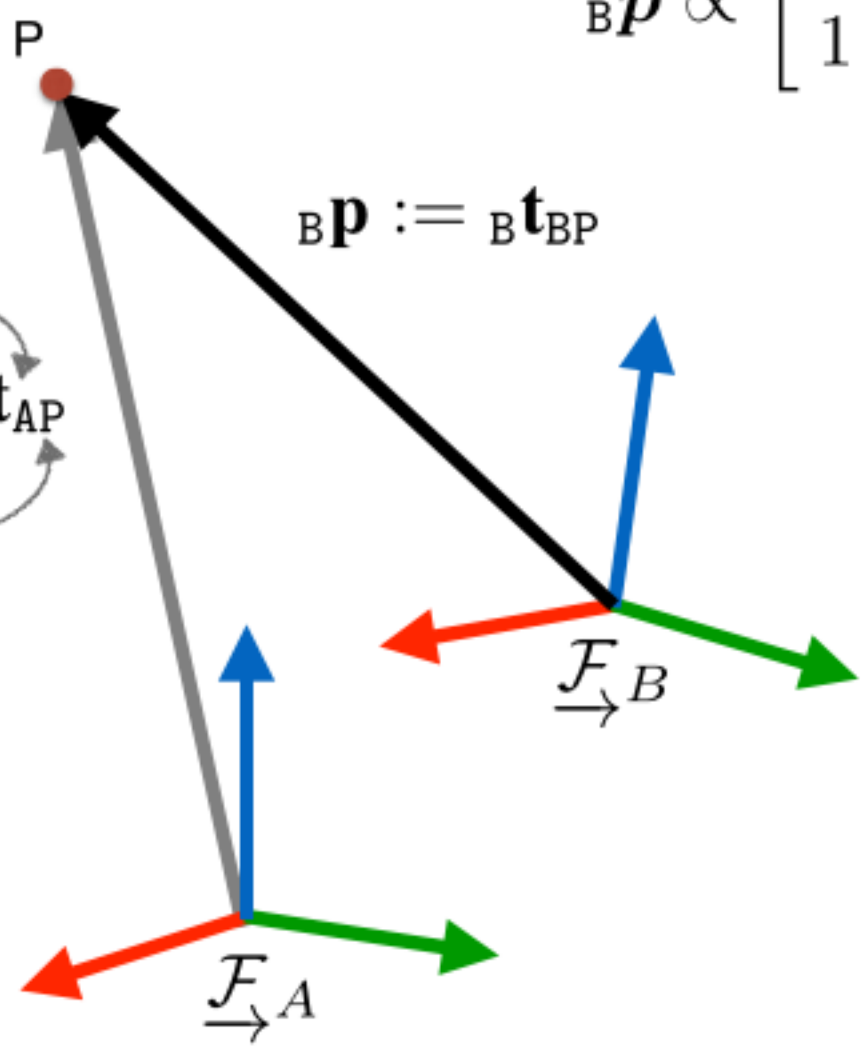
The Ugly: Notation

$$T_{\mathtt{AB}}$$

into — transforms points from

$$_{\mathtt{B}}\boldsymbol{p} \propto \begin{bmatrix} _{\mathtt{B}}\mathbf{p} \\ 1 \end{bmatrix}$$

$$_{\mathtt{A}}\mathbf{p} := {_{\mathtt{A}}}\mathbf{t}_{\mathtt{AP}}$$



$$_{\mathtt{A}}\boldsymbol{p} = \boldsymbol{T}_{\mathtt{AB}}\,{_{\mathtt{B}}}\boldsymbol{p}$$

The Ugly: Notation

$$T_{\mathtt{AB}}$$

With respect to

of

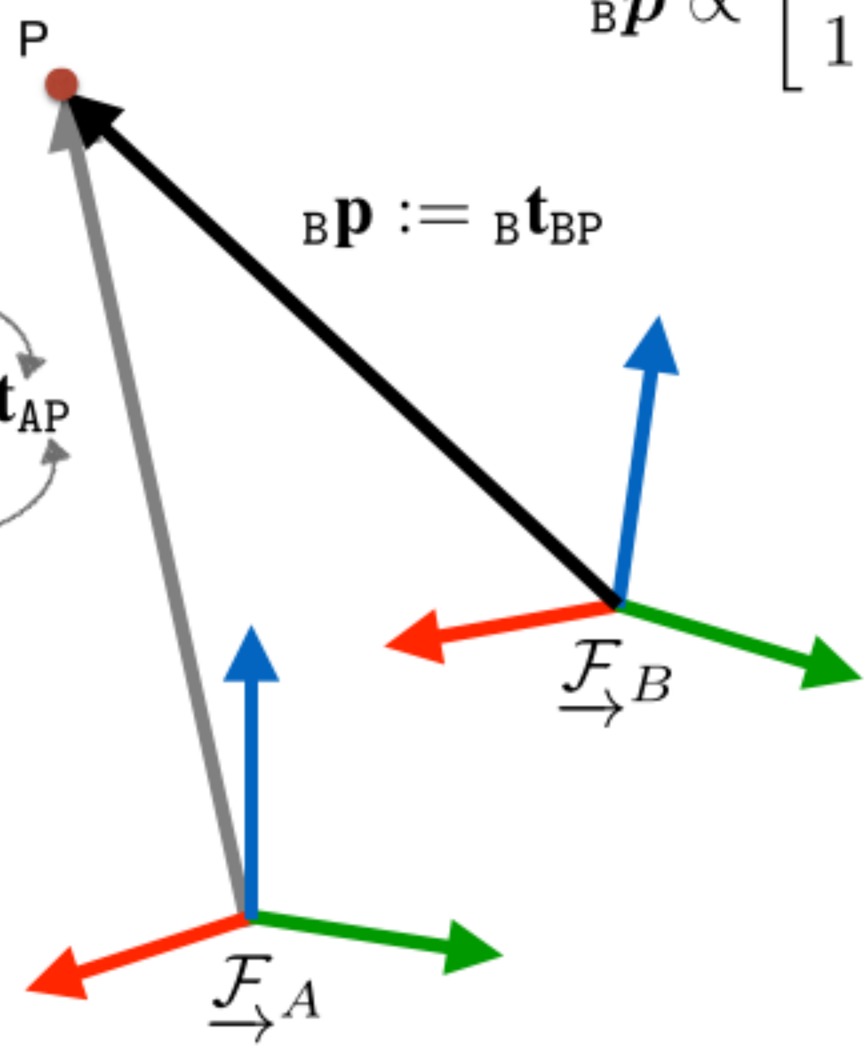
$${}_{\mathtt{A}}\boldsymbol{p} = T_{\mathtt{AB}}\,{}_{\mathtt{B}}\boldsymbol{p}$$

"The pose of B with respect to A"

$${}_{\mathtt{B}}\boldsymbol{p} \propto \begin{bmatrix} {}_{\mathtt{B}}\mathbf{p} \\ 1 \end{bmatrix}$$

$${}_{\mathtt{A}}\mathbf{p} := {}_{\mathtt{A}}\mathbf{t}_{\mathtt{AP}}$$

$${}_{\mathtt{B}}\mathbf{p} := {}_{\mathtt{B}}\mathbf{t}_{\mathtt{BP}}$$



Recommendation 5

Choose expressive notation.

Explain it clearly.

Stick with it.

Let’s talk about code.

The Ugly: Code

- Code has the same requirements as notation
- Rotation matrices have two frame decorations:
  - to
  - from
- Coordinates of vectors have three decorations:
  - to
  - from
  - expressed in

The Ugly: Code

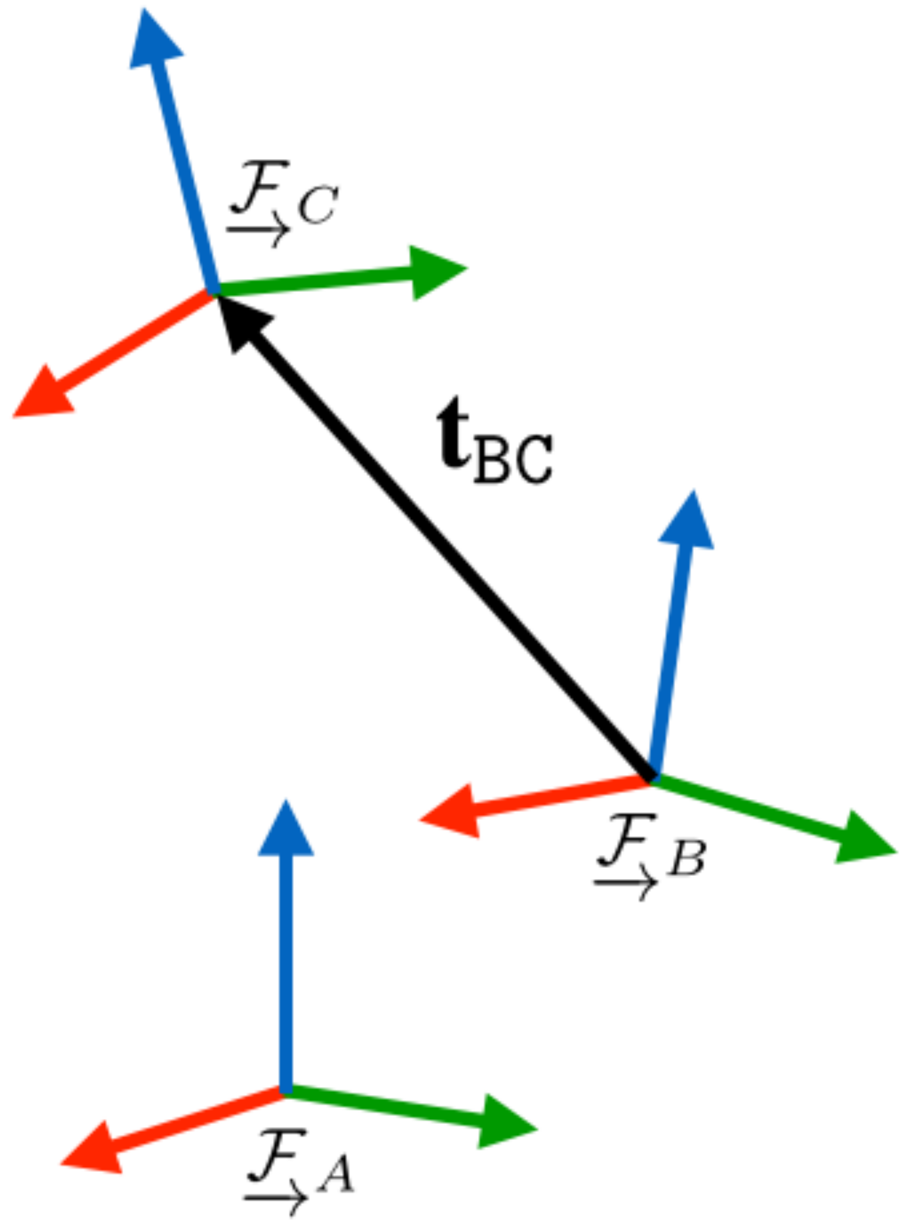
Physical quantity

$${}_{A}\mathbf{t}_{BC}$$

Expressed in

With respect to

of



The Ugly: Code

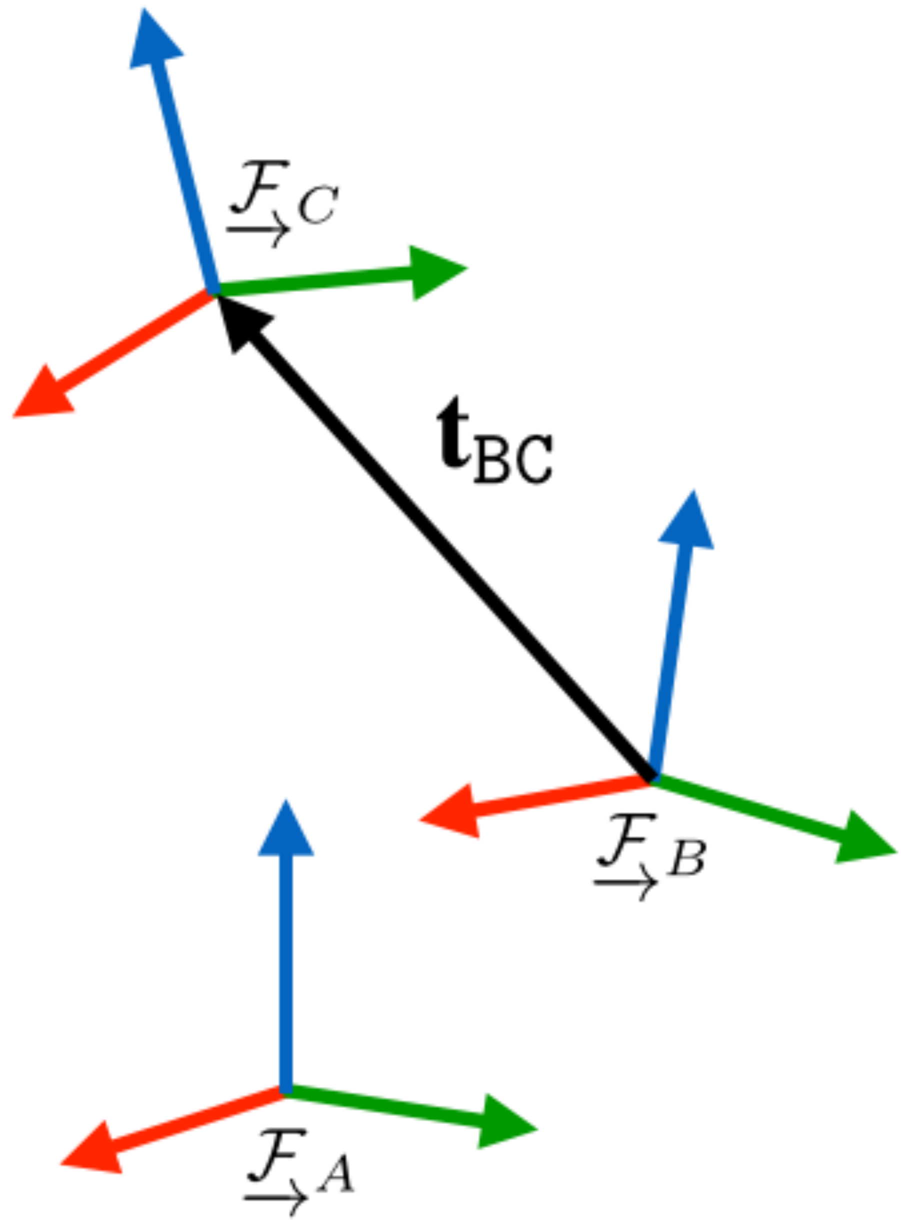
Physical quantity

`A_t_B_C`

Expressed in

With respect to

of



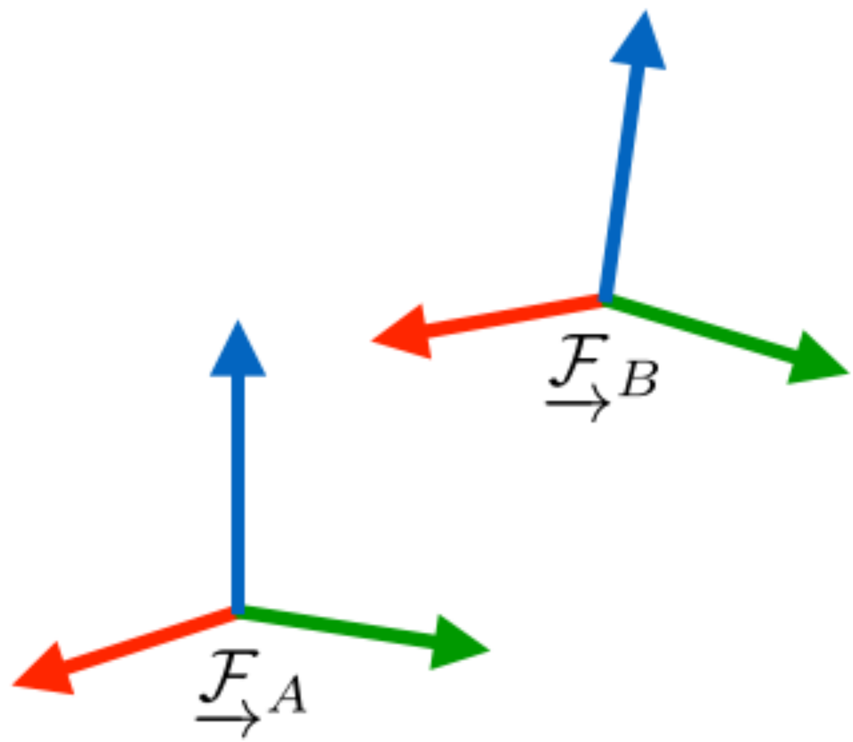
The Ugly: Code

$$\mathbf{C}_{\mathtt{AB}}$$

into

rotates vectors from

$$_{\mathtt{A}}\mathbf{v}_{\mathtt{BC}} = \mathbf{C}_{\mathtt{AB}}\,{}_{\mathtt{B}}\mathbf{v}_{\mathtt{BC}}$$



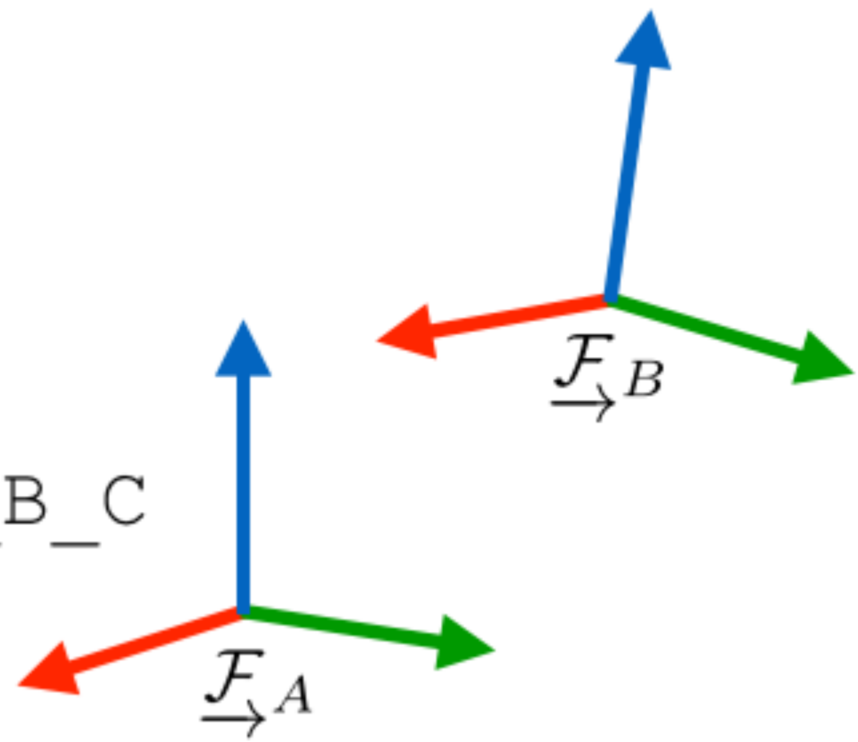
The Ugly: Code

```
C_A_B
```

into

rotates vectors from

```
A_v_B_C = C_A_B * B_v_B_C
```



The Ugly: Code

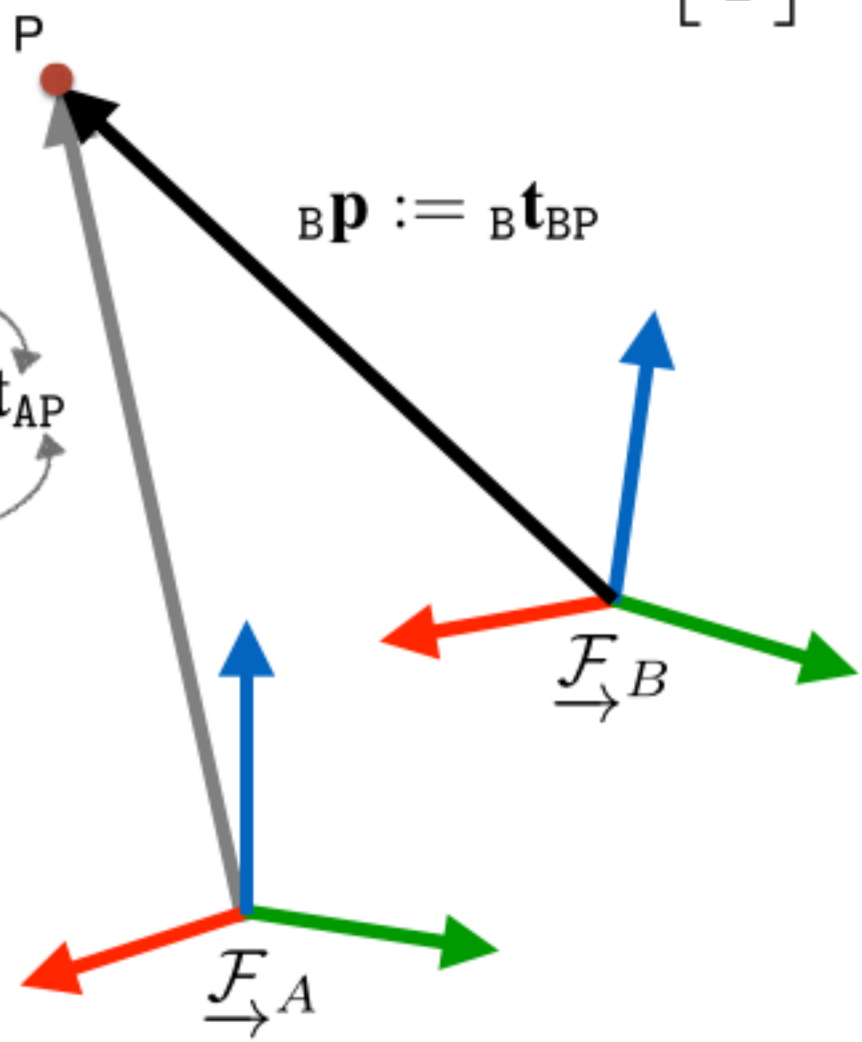
$$_{\mathrm{B}}\boldsymbol{p} \propto \begin{bmatrix} _{\mathrm{B}}\mathbf{p} \\ 1 \end{bmatrix}$$

$$\boldsymbol{T}_{\mathrm{AB}}$$

into

transforms
points
from

$$_{\mathrm{A}}\mathbf{p} := {_{\mathrm{A}}\mathbf{t}_{\mathrm{AP}}}$$



$$_{\mathrm{A}}\boldsymbol{p} = \boldsymbol{T}_{\mathrm{AB}\,\mathrm{B}}\boldsymbol{p}$$

The Ugly: Code

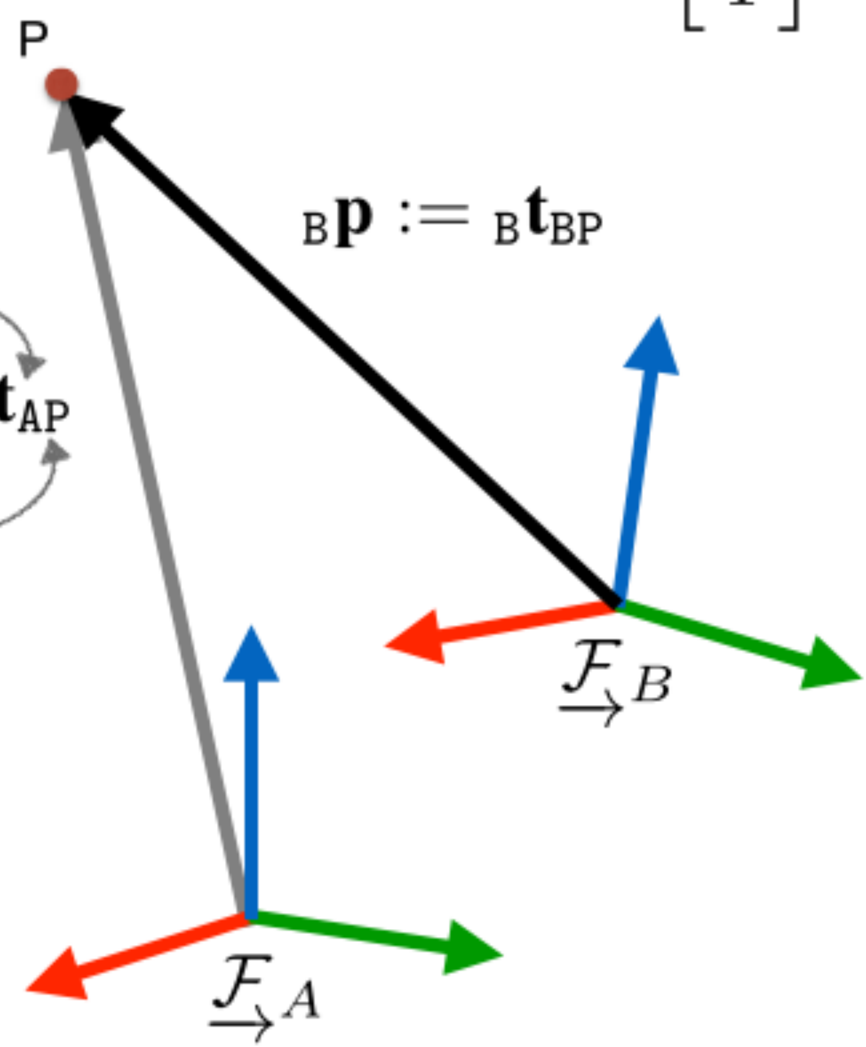
$$ {}_{\mathrm{B}}\boldsymbol{p} \propto \begin{bmatrix} {}_{\mathrm{B}}\mathbf{p} \\ 1 \end{bmatrix} $$

```
T_A_B        A_p
```

into

transforms points from

$$ {}_{\mathrm{A}}\mathbf{p} := {}_{\mathrm{A}}\mathbf{t}_{\mathrm{AP}} $$




```
A_p = T_A_B * B_p
```

The Ugly: Code

- Comments

```
/// Coordinate frames in this function:
///   - C : The camera frame, indexed by time, k.
///   - W : The world frame.
Point pointToCamera( const Transformation& T_W_Ckm1,
                     const Transformation& T_Ckm1_Ck,
                     const Transformation& T_Ck_Ckp1,
                     const Point& W_p ) {

  Transformation T_Ckp1_W = (T_W_Ckm1 * T_Ckm1_Ck * T_Ck_Ckp1).inverse();
  return T_Ckp1_W * W_p

}
```

The Ugly: Code

- Comments

$T_{\mathtt{WC}_{\mathtt{k-1}}}$

```
/// Coordinate frames in this function:
///   - C : The camera frame, indexed by time, k.
///   - W : The world frame.
Point pointToCamera( const Transformation& T_W_Ckm1,
                     const Transformation& T_Ckm1_Ck,
                     const Transformation& T_Ck_Ckp1,
                     const Point& W_p ) {

  Transformation T_Ckp1_W = (T_W_Ckm1 * T_Ckm1_Ck * T_Ck_Ckp1).inverse();
  return T_Ckp1_W * W_p

}
```

The Ugly: Code

- Comments

$T_{\mathtt{WC}_{\mathtt{k-1}}}$

$T_{\mathtt{C}_{\mathtt{k-1}}\mathtt{C}_{\mathtt{k}}}$

```
/// Coordinate frames in this function:
///   - C : The camera frame, indexed by time, k.
///   - W : The world frame.
Point pointToCamera( const Transformation& T_W_Ckm1,
                     const Transformation& T_Ckm1_Ck,
                     const Transformation& T_Ck_Ckp1,
                     const Point& W_p ) {

  Transformation T_Ckp1_W = (T_W_Ckm1 * T_Ckm1_Ck * T_Ck_Ckp1).inverse();
  return T_Ckp1_W * W_p

}
```

The Ugly: Code

- Comments

$\boldsymbol{T}_{\mathtt{WC_{k-1}}}$

$\boldsymbol{T}_{\mathtt{C_{k-1}C_k}}$

$\boldsymbol{T}_{\mathtt{C_kC_{k+1}}}$

```
/// Coordinate frames in this function:
///   - C : The camera frame, indexed by time, k.
///   - W : The world frame.
Point pointToCamera( const Transformation& T_W_Ckm1,
                     const Transformation& T_Ckm1_Ck,
                     const Transformation& T_Ck_Ckp1,
                     const Point& W_p ) {

  Transformation T_Ckp1_W = (T_W_Ckm1 * T_Ckm1_Ck * T_Ck_Ckp1).inverse();
  return T_Ckp1_W * W_p

}
```

The Ugly: Code

- Comments

```
/// Coordinate frames in this function:
///   - C : The camera frame, indexed by time, k.
///   - W : The world frame.
Point pointToCamera( const Transformation& T_W_Ckm1,
                     const Transformation& T_Ckm1_Ck,
                     const Transformation& T_Ck_Ckp1,
                     const Point& W_p ) {

  Transformation T_Ckp1_W = (T_W_Ckm1 * T_Ckm1_Ck * T_Ck_Ckp1).inverse();
  return T_Ckp1_W * W_p

}
```

$\boldsymbol{T}_{\mathtt{W}\mathtt{C}_{\mathtt{k}-1}}$

$\boldsymbol{T}_{\mathtt{C}_{\mathtt{k}-1}\mathtt{C}_{\mathtt{k}}}$

$\boldsymbol{T}_{\mathtt{C}_{\mathtt{k}}\mathtt{C}_{\mathtt{k}+1}}$

${}_{\mathtt{W}}\mathbf{p}$

Recommendation 6

Choose an expressive coding style.

Explain it clearly.

Stick with it.

Summary

Goal: Reduce the amount of suffering in the world.

Recommendation 1

Have fear.

Recommendation 2

Always provide a frame diagram.

Recommendation 3

Be clear about your orientation representation.

1. How to build a rotation matrix.
2. What rotation matrix that is.

Recommendation 4

Be clear about your pose representation.

1. How to build a transformation matrix.
2. What transformation matrix that is.

Recommendation 5

Choose expressive notation.
Explain it clearly.
Stick with it.

Recommendation 6

Choose an expressive coding style.

Explain it clearly.

Stick with it.

Fork me on Github

Kindr: Kinematics and Dynamics for Robotics

- Full suite of unit tests
- Online documentation (down to the scalar level)
- Recommended coding style
- Notation and concept cheat sheet
- https://github.com/ethz-asl/kindr
- Thanks: Michael Bloesch, Remo Diethelm, Peter Fankhauser, Christian Gehring, Hannes Sommer

Kindr Cheat Sheet

Kinematics and Dynamics for Robotics

Nomenclature

(Hyper-)complex number	Q	normal capital letter
Column vector	$\mathbf{a}$	bold small letter
Matrix	$\mathbf{M}$	bold capital letter
Identity matrix	$\mathbb{1}_{n\times m}$	$n \times m$-matrix
Coordinate system (CS)	$\mathbf{e}_x^A, \mathbf{e}_y^A, \mathbf{e}_z^A$	Cartesian right-hand system A with basis (unit) vectors $\mathbf{e}$
Inertial frame	$\mathbf{e}_x^I, \mathbf{e}_y^I, \mathbf{e}_z^I$	Global / inertial / world coordinate system (never moves)
Body-fixed frame	$\mathbf{e}_x^B, \mathbf{e}_y^B, \mathbf{e}_z^B$	Local / body-fixed coordinate system (moves with body)
Rotation	$\mathcal{R} \in SO(3)$	generic rotation (for all parameterizations)
Machine precision	ϵ	

Operators

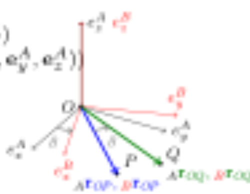
Cross product	$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \Leftrightarrow (\mathbf{a})^\wedge \mathbf{b} = \hat{\mathbf{a}}\mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$
Skew/unskew	$\mathbf{a} = \hat{\mathbf{a}}^\vee$
Euclidean norm	$\Vert\mathbf{a}\Vert = \sqrt{\mathbf{a}^T\mathbf{a}} = \sqrt{a_1^2 + \ldots + a_n^2}$
Exponential map for matrix	$\mathrm{expM} : \mathbb{R}^3 \to \mathbb{R}^3, \mathbf{A} \mapsto e^{\mathbf{A}}, \quad \mathbf{A} \in \mathbb{R}^{3\times 3}$
Logarithmic map for matrix	$\mathrm{logM} : \mathbb{R}^3 \to \mathbb{R}^3, \mathbf{A} \mapsto \log \mathbf{A}, \quad \mathbf{A} \in \mathbb{R}^{3\times 3}$

Position & Orientation

Position

Vector	$\mathbf{r}_{OP}$	from point O to point P
Position vector	${}_A\mathbf{r}_{OP} \in \mathbb{R}^3$	from point O to point P expr. in frame A
Homogeneous pos. vector	${}_A\bar{\mathbf{r}}_{OP} = \begin{bmatrix} {}_A\mathbf{r}_{OP}^T & 1 \end{bmatrix}^T$	from point O to point P expr. in frame A

Orientation/Rotation



1) Active Rotation: $\mathcal{R}^A : {}_A\mathbf{r}_{OP} \mapsto {}_A\mathbf{r}_{OQ}$ (rotates the vector $\mathbf{r}_{OP}$)
2) Passive Rotation: $\mathcal{R}^P : {}_A\mathbf{r}_{OP} \mapsto {}_B\mathbf{r}_{OP}$ (rotates the frame $(\mathbf{e}_x^A, \mathbf{e}_y^A, \mathbf{e}_z^A)$)
3) Inversion: $\mathcal{R}^{A^{-1}}(\mathbf{r}) = \mathcal{R}^P(\mathbf{r})$
4) Concatenation:
$$\mathcal{R}_2^A\left(\mathcal{R}_1^A(\mathbf{r})\right) = \left(\mathcal{R}_2^A \otimes \mathcal{R}_1^A\right)(\mathbf{r}) = \left(\mathcal{R}_1^{A^{-1}} \otimes \mathcal{R}_2^{A^{-1}}\right)^{-1}(\mathbf{r})$$
$$\mathcal{R}_2^P\left(\mathcal{R}_1^P(\mathbf{r})\right) = \left(\mathcal{R}_2^P \otimes \mathcal{R}_1^P\right)(\mathbf{r}) = \left(\mathcal{R}_1^{P^{-1}} \otimes \mathcal{R}_2^{P^{-1}}\right)^{-1}(\mathbf{r})$$
5) Exponential map: $\exp : \mathbb{R}^3 \to SO(3), \mathbf{v} \mapsto \mathrm{expM}(\hat{\mathbf{v}}), \quad \mathbf{v} \in \mathbb{R}^3$
6) Logarithmic map: $\log : SO(3) \to \mathbb{R}^3, \mathcal{R} \mapsto \mathrm{logM}(\mathcal{R})^\vee, \quad \mathcal{R} \in SO(3)$
7) Box plus: $\mathcal{R}_2 = \mathcal{R}_1 \boxplus \mathbf{v} = \exp(\mathbf{v}) \otimes \mathcal{R}_1, \quad \mathcal{R}_1, \mathcal{R}_2 \in SO(3), \mathbf{v} \in \mathbb{R}^3$
8) Box minus: $\mathbf{v} = \mathcal{R}_1 \boxminus \mathcal{R}_2 = \log(\mathcal{R}_1 \otimes \mathcal{R}_2^{-1}), \quad \mathcal{R}_1, \mathcal{R}_2 \in SO(3), \mathbf{v} \in \mathbb{R}^3$
9) Discrete integration: $\mathcal{R}^{k+1} = \mathcal{R}^k \boxplus ({}_B\boldsymbol{\omega}_{IB}\Delta t)$
10) (Spherical) linear interpolation $t \in [0,1]$: $\mathcal{R}_t = \mathcal{R}_0 \boxplus ((\mathcal{R}_1 \boxminus \mathcal{R}_0)t), \quad \mathcal{R}_t = \mathcal{R}(t), \mathcal{R}_0 = \mathcal{R}(0), \mathcal{R}_1 = \mathcal{R}(1)$
$$= (\mathcal{R}_1 \otimes \mathcal{R}_0^{-1})^t \otimes \mathcal{R}_0$$

Rotation Matrix	$\mathbf{R}_{BA} \in SO(3)$ ${}_A\mathbf{r}_{OQ} = \mathbf{R}_{BA}\,{}_A\mathbf{r}_{OP}$ ${}_B\mathbf{r}_{OP} = \mathbf{R}_{BA}^T\,{}_A\mathbf{r}_{OP}$	Maps the coord. of the basis vectors $({}_A\mathbf{e}_x^A, {}_A\mathbf{e}_y^A, {}_A\mathbf{e}_z^A)$ of frame A expressed in A into the coordinates of the basis vectors $({}_A\mathbf{e}_x^B, {}_A\mathbf{e}_y^B, {}_A\mathbf{e}_z^B)$ of B expressed in A. The rotation is active (alibi). ${}_A\mathbf{R}_{BA} = \begin{bmatrix} {}_A\mathbf{e}_x^B & {}_A\mathbf{e}_y^B & {}_A\mathbf{e}_z^B \end{bmatrix}$
Direct Cosine Matrix	$\mathbf{C}_{BA} \in SO(3)$ ${}_B\mathbf{r}_{OP} = \mathbf{C}_{BA}\,{}_A\mathbf{r}_{OP}$ $\mathbf{C}_{BA} = \mathbf{R}_{BA}^T$	The coordinate tranformation matrix, which transforms vectors from frame A to frame B. The rotation is passive (alias).
Rotation Quaternion	$\mathbf{p}_{BA}$ $\mathbf{p}_{BA} \Leftrightarrow \mathbf{R}_{BA}$	The rotation is active (alibi).
Rotation Angle-axis	$(\theta, \mathbf{n})_{BA}$ $(\theta, \mathbf{n})_{BA} \Leftrightarrow \mathbf{R}_{BA}$	Rotation with unit rotation axis $\mathbf{n}$ and angle $\theta \in [0, \pi]$. The rotation is active (alibi).
Rotation Vector	$\boldsymbol{\phi}_{BA}$ $\boldsymbol{\phi}_{BA} \Leftrightarrow \mathbf{R}_{BA}$	Rotation with rotation axis $\mathbf{n} = \frac{\boldsymbol{\phi}}{\Vert\boldsymbol{\phi}\Vert}$ and angle $\theta = \Vert\boldsymbol{\phi}\Vert$. The rotation is active (alibi).
Euler Angles ZYX Euler Angles YPR	$(\psi, \theta, \phi)_{BA}$ $(\psi, \theta, \phi)_{BA} \Leftrightarrow \mathbf{C}_{BA}$	Tait-Bryan angles (Flight conv.): $z - y' - x''$, i.e. yaw-pitch-roll. Singularities are at $\theta = \pm\frac{\pi}{2}$. $\psi \in [-\pi, \pi), \theta \in [-\frac{\pi}{2}, \frac{\pi}{2}), \phi \in [-\pi, \pi)$
Euler Angles XYZ Euler Angles RPY	$(\alpha, \beta, \gamma)_{BA}$ $(\alpha, \beta, \gamma)_{BA} \Leftrightarrow \mathbf{C}_{BA}$	Cardan angles (Glocker conv.): $x - y' - z''$, i.e. roll-pitch-yaw. Singularities are at $\beta = \pm\frac{\pi}{2}$. $\alpha \in [-\pi, \pi), \beta \in [-\frac{\pi}{2}, \frac{\pi}{2}), \gamma \in [-\pi, \pi)$

Rotation Quaternion

A rotation quaternion is a Hamiltonian unit quaternion:

$$\boxed{\begin{aligned} &P = p_0 + p_1 i + p_2 j + p_3 k \in \mathbb{H}, \quad p_i \in \mathbb{R} \\ &i^2 = j^2 = k^2 = ijk = -1, \quad \Vert P\Vert = \sqrt{p_0^2 + p_1^2 + p_2^2 + p_3^2} = 1 \end{aligned}}$$

Note that P_{BA} and $-P_{BA}$ represent the same rotation, but not the same unit quaternion.
Rot. quaternion as tuple: $P = (p_0, p_1, p_2, p_3) = (p_0, \vec{\mathbf{p}})$ with $\vec{\mathbf{p}} := (p_1, p_2, p_3)^T$
Rot. quaternion as vector: $\mathbf{p} = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 \end{bmatrix}^T$
Conjugate: $P^* = (p_0, -\vec{\mathbf{p}})$
Inverse: $P^{-1} = P^* = (p_0, -\vec{\mathbf{p}})$
Quaternion multiplication:
$$Q \cdot P = (q_0, \vec{\mathbf{q}}) \cdot (p_0, \vec{\mathbf{p}}) = (q_0 p_0 - \vec{\mathbf{q}}^T\vec{\mathbf{p}}, q_0\vec{\mathbf{p}} + p_0\vec{\mathbf{q}} + \vec{\mathbf{q}} \times \vec{\mathbf{p}}) \quad \Leftrightarrow$$
$$\mathbf{q} \otimes \mathbf{p} = \underbrace{\mathbf{Q}(\mathbf{q})}_{\text{quaternion matrix}}\mathbf{p} = \begin{pmatrix} q_0 & -\vec{\mathbf{q}}^T \\ \vec{\mathbf{q}} & q_0 \mathbb{1}_{3\times 3} + \hat{\vec{\mathbf{q}}} \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix} = \begin{pmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$$
$$= \underbrace{\bar{\mathbf{Q}}(\mathbf{p})}_{\text{conjugate quat. matrix}}\mathbf{q} = \begin{pmatrix} p_0 & -\vec{\mathbf{p}}^T \\ \vec{\mathbf{p}} & p_0 \mathbb{1}_{3\times 3} - \hat{\vec{\mathbf{p}}} \end{pmatrix} \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & p_3 & -p_2 \\ p_2 & -p_3 & p_0 & p_1 \\ p_3 & p_2 & -p_1 & p_0 \end{pmatrix} \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix}$$

Rotation Quaternion ⇔ Rotation Angle-Axis

$$\mathbf{p}_{BI} = \begin{bmatrix} \cos\frac{\theta}{2} \\ \mathbf{n}\sin\frac{\theta}{2} \end{bmatrix} \quad \Leftrightarrow \quad (\theta, \mathbf{n})_{BI} = \begin{cases} \left(2\arccos(p_0), \frac{\vec{\mathbf{p}}}{\Vert\vec{\mathbf{p}}\Vert}\right) & \text{if } \Vert\vec{\mathbf{p}}\Vert^2 \geq \epsilon^2 \\ \left(0, \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T\right) & \text{otherwise} \end{cases}$$

Rotation Quaternion ⇔ Direction Cosine Matrix

$$\mathbf{C}_{AB} = \mathbf{R}_{AB}^T(\mathbf{p}_{AB}) = \mathbb{1}_{3\times 3} + 2p_0\hat{\vec{\mathbf{p}}} + 2\hat{\vec{\mathbf{p}}}^2$$
$$= \begin{bmatrix} p_0^2 + p_1^2 - p_2^2 - p_3^2 & 2p_1p_2 - 2p_0p_3 & 2p_0p_2 + 2p_1p_3 \\ 2p_0p_3 + 2p_1p_2 & p_0^2 - p_1^2 + p_2^2 - p_3^2 & 2p_2p_3 - 2p_0p_1 \\ 2p_1p_3 - 2p_0p_2 & 2p_0p_1 + 2p_2p_3 & p_0^2 - p_1^2 - p_2^2 + p_3^2 \end{bmatrix}$$