Contents

Numerical Comparison

In this section, Euler approach and Dual Quaternion approach are compared and stored into a particular structure. One example is also shown for getting values.

```
clear all;
clc;
close all;
```

Loading DH Parameters

```
load('DH_para_robots.mat');
```

KR3 Euler

```
        % Elbow Non-Singuler
        kr3.Euler.Elbow.Non_Singular_Theta = [deg2rad(0) deg2rad(30) deg2rad(-45) deg2rad(80) deg2rad(30) deg2rad(30)];
        kr3.Euler.Elbow.Non_Singular = KR3_Eular(kr3.Euler.Elbow.Non_Singular_Theta,KR3);

        % Elbow Singuler
        kr3.Euler.Elbow.Singular_Theta = [deg2rad(0) deg2rad(49.5) deg2rad(-4.5) deg2rad(80) deg2rad(30) deg2rad(30)];
        kr3.Euler.Elbow.Singular = KR3_Eular(kr3.Euler.Elbow.Singular_Theta,KR3);

        % Wrist Non-Singuler
        kr3.Euler.Wrist.Non_Singular_Theta = [deg2rad(0) deg2rad(30) deg2rad(-45) deg2rad(80) deg2rad(30) deg2rad(30)];
        kr3.Euler.Wrist.Non_Singular = KR3_Eular(kr3.Euler.Wrist.Non_Singular_Theta,KR3);

        % wrist Singuler
        kr3.Euler.Wrist.Singular_Theta = [deg2rad(0) deg2rad(30) deg2rad(-45) deg2rad(80) deg2rad(0) deg2rad(30)];
        kr3.Euler.Wrist.Singular = KR3_Eular(kr3.Euler.Wrist.Singular_Theta,KR3);
```

KR3 Dual Quaternion

```
        % Elbow Non-Singuler
        kr3.Dual_Quat.Elbow.Non_Singular_Theta = kr3.Euler.Elbow.Non_Singular_Theta;
        kr3.Dual_Quat.Elbow.Non_Singular = KR3_Quaternion(kr3.Dual_Quat.Elbow.Non_Singular_Theta);

        % Elbow Singuler
        kr3.Dual_Quat.Elbow.Singular_Theta = kr3.Euler.Elbow.Singular_Theta;
        kr3.Dual_Quat.Elbow.Singular = KR3_Quaternion(kr3.Dual_Quat.Elbow.Singular_Theta);

        % Wrist Non-Singuler
        kr3.Dual_Quat.Wrist.Non_Singular_Theta = kr3.Euler.Wrist.Non_Singular_Theta;
        kr3.Dual_Quat.Wrist.Non_Singular = KR3_Quaternion(kr3.Dual_Quat.Wrist.Non_Singular_Theta);

        % Wrist Singuler
        kr3.Dual_Quat.Wrist.Singular_Theta = kr3.Euler.Wrist.Singular_Theta;
        kr3.Dual_Quat.Wrist.Singular = KR3_Quaternion(kr3.Dual_Quat.Wrist.Singular_Theta);

        %Example
        kr3
        kr3.Euler
        kr3.Euler.Elbow
        kr3.Euler.Elbow.Non_Singular
        kr3.Euler.Elbow.Non_Singular.jcob
```

```
kr3 = 

  struct with fields:

        Euler: [1×1 struct]
    Dual_Quat: [1×1 struct]


ans = 

  struct with fields:

    Elbow: [1×1 struct]
    Wrist: [1×1 struct]


ans = 

  struct with fields:

    Non_Singular_Theta: [0 0.5236 -0.7854 1.3963 0.5236 0.5236]
          Non_Singular: [1×1 struct]
        Singular_Theta: [0 0.8639 -0.0785 1.3963 0.5236 0.5236]
              Singular: [1×1 struct]


ans = 

  struct with fields:

     jcob: [6×6 double]
     jdet: -0.0111
    jcond: 24.7028
    jrank: 6
    trans: [3×6 double]
     tdet: 8.2724e-04
    tcond: 8.5143
    trank: 3
      rot: [3×6 double]
     rdet: 6.3481
    rcond: 1.6265
    rrank: 3


ans =

    0.0000   -0.1014    0.0286   -0.0000   -0.0194         0
    0.5739   -0.0000   -0.0000   -0.0375   -0.0000         0
   -0.0000    0.5539    0.3288    0.0000    0.0724         0
    0.0000   -0.0000   -0.0000    0.9659   -0.0000    0.9659
   -0.0000   -1.0000   -1.0000   -0.0000   -1.0000   -0.0000
    1.0000    0.0000    0.0000   -0.2588    0.0000    0.2588
```

ABB IRB140 Euler

```
        % Elbow Non-Singuler
        abb.Euler.Elbow.Non_Singular_Theta = [deg2rad(0) deg2rad(30) deg2rad(-45) deg2rad(80) deg2rad(30) deg2rad(30)];
        abb.Euler.Elbow.Non_Singular = ABB_Eular(abb.Euler.Elbow.Non_Singular_Theta,irb140);

        % Elbow Singuler
        abb.Euler.Elbow.Singular_Theta = [deg2rad(0) deg2rad(49.5) deg2rad(-4.5) deg2rad(80) deg2rad(30) deg2rad(30)];
        abb.Euler.Elbow.Singular = ABB_Eular(abb.Euler.Elbow.Singular_Theta,irb140);

        % Wrist Non-Singuler
        abb.Euler.Wrist.Non_Singular_Theta = [deg2rad(0) deg2rad(30) deg2rad(-45) deg2rad(80) deg2rad(30) deg2rad(30)];
        abb.Euler.Wrist.Non_Singular = ABB_Eular(abb.Euler.Wrist.Non_Singular_Theta,irb140);
        % wrist Singuler
        abb.Euler.Wrist.Singular_Theta = [deg2rad(0) deg2rad(30) deg2rad(-45) deg2rad(80) deg2rad(0) deg2rad(30)];
        abb.Euler.Wrist.Singular = ABB_Eular(abb.Euler.Wrist.Singular_Theta,irb140);
```

ABB IRB140 Dual Quaternion

```
        % Elbow Non-Singuler
        abb.Dual_Quat.Elbow.Non_Singular_Theta = abb.Euler.Elbow.Non_Singular_Theta;
        abb.Dual_Quat.Elbow.Non_Singular = ABB_Quaternion(abb.Dual_Quat.Elbow.Non_Singular_Theta);

        % Elbow Singuler
        abb.Dual_Quat.Elbow.Singular_Theta = abb.Euler.Elbow.Singular_Theta;
        abb.Dual_Quat.Elbow.Singular = ABB_Quaternion(abb.Dual_Quat.Elbow.Singular_Theta);

        % Wrist Non-Singuler
        abb.Dual_Quat.Wrist.Non_Singular_Theta = abb.Euler.Wrist.Non_Singular_Theta;
        abb.Dual_Quat.Wrist.Non_Singular = ABB_Quaternion(abb.Dual_Quat.Wrist.Non_Singular_Theta);

        % Wrist Singuler
        abb.Dual_Quat.Wrist.Singular_Theta = abb.Euler.Wrist.Singular_Theta;
        abb.Dual_Quat.Wrist.Singular = ABB_Quaternion(abb.Dual_Quat.Wrist.Singular_Theta);
```

Mitsubishi RV-2AJ Euler

```
        % Elbow Non-Singuler
        rv2aj.Euler.Elbow.Non_Singular_Theta = [deg2rad(0) deg2rad(45) deg2rad(60) deg2rad(30) deg2rad(30)];
        rv2aj.Euler.Elbow.Non_Singular = RV2_Eular(rv2aj.Euler.Elbow.Non_Singular_Theta,rv2);

        % Elbow Singuler
        rv2aj.Euler.Elbow.Singular_Theta = [deg2rad(0) deg2rad(45) deg2rad(0) deg2rad(30) deg2rad(30)];
        rv2aj.Euler.Elbow.Singular = RV2_Eular(rv2aj.Euler.Elbow.Singular_Theta,rv2);

        % Wrist Non-Singuler
        rv2aj.Euler.Wrist.Non_Singular_Theta = [deg2rad(0) deg2rad(45) deg2rad(60) deg2rad(30) deg2rad(30)];
        rv2aj.Euler.Wrist.Non_Singular = RV2_Eular(rv2aj.Euler.Wrist.Non_Singular_Theta,rv2);
```

Mitsubishi RV-2AJ Dual Quaternion

```
        % Elbow Non-Singuler
        rv2aj.Dual_Quat.Elbow.Non_Singular_Theta = rv2aj.Euler.Elbow.Non_Singular_Theta;
        rv2aj.Dual_Quat.Elbow.Non_Singular = RV2_Quaternion(rv2aj.Dual_Quat.Elbow.Non_Singular_Theta);

        % Elbow Singuler
        rv2aj.Dual_Quat.Elbow.Singular_Theta = rv2aj.Euler.Elbow.Singular_Theta;
        rv2aj.Dual_Quat.Elbow.Singular = RV2_Quaternion(rv2aj.Dual_Quat.Elbow.Singular_Theta);

        % Wrist Non-Singuler
        rv2aj.Dual_Quat.Wrist.Non_Singular_Theta = rv2aj.Euler.Wrist.Non_Singular_Theta;
        rv2aj.Dual_Quat.Wrist.Non_Singular = RV2_Quaternion(rv2aj.Dual_Quat.Wrist.Non_Singular_Theta);
```

Published with MATLAB® R2019b