Inverse Kinematics Solver for 6 Degree of Freedom Robotic Manipulator

Generated by Doxygen 1.8.17

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 Forward_Kinematics Class Reference

The following Class contains all the methods,attributes of Forward Kinematics Class. It provide methods to solve the forward kinematics of a robotic manipulator.

`#include <Forward_kinematics.hpp>`

Public Member Functions

- void solve_FK (const std::vector< double > &_input_joint_angles)

  this function will calculate the end effector position rom the given input_joint_angles.
- void set_output_coordinates (const std::vector< double > &_output_joint_coordinates)

  It sets the output_coordinates(input) to the output_joint_coordinates.
- void set_output_angles (const std::vector< double > &_output_joint_angles)

  It sets the given input to output_joint_coordinates.
- void set_input_angles (const std::vector< double > &_input_joint_angles)

  It sets the given input to input_joint_angles.
- void set_current_pose (const std::vector< double > &_current_robot_pose)

  It sets the given input to current_robot_pose.
- std::vector< double > get_output_coordinates ()

  Getter method for returning output_joint_coordinates.
- std::vector< double > get_output_angles ()

  Getter Method for returning output_joint_angles.
- std::vector< double > get_current_pose ()

  Getter method for returning the current_robot_pose.
- std::vector< double > get_input_angles ()

  Getter method for getting the input_joint_angles.

3.1.1 Detailed Description

The following Class contains all the methods,attributes of Forward Kinematics Class. It provide methods to solve the forward kinematics of a robotic manipulator.

3.1.2 Member Function Documentation

3.1.2.1 get_current_pose()

```
std::vector< double > Forward_Kinematics::get_current_pose ( )
```

Getter method for returning the current_robot_pose.

Returns

current_robot_pose

3.1.2.2 get_input_angles()

```
std::vector< double > Forward_Kinematics::get_input_angles ( )
```

Getter method for getting the input_joint_angles.

Returns

input_joint_angles

3.1.2.3 get_output_angles()

```
std::vector< double > Forward_Kinematics::get_output_angles ( )
```

Getter Method for returning output_joint_angles.

Returns

output_joint_angles

3.1.2.4 get_output_coordinates()

```
std::vector< double > Forward_Kinematics::get_output_coordinates ( )
```

Getter method for returning output_joint_coordinates.

Returns

output_joint_coordinates

3.1.2.5 set_current_pose()

```
void Forward_Kinematics::set_current_pose (
            const std::vector< double > & _current_robot_pose )
```

It sets the given input to current_robot_pose.

Parameters

_current_robot_pose	

Returns

None

3.1.2.6 set_input_angles()

```
void Forward_Kinematics::set_input_angles (
            const std::vector< double > & _input_joint_angles )
```

It sets the given input to input_joint_angles.

Parameters

_input_joint_angles	

Returns

None

3.1.2.7 set_output_angles()

```
void Forward_Kinematics::set_output_angles (
            const std::vector< double > & _output_joint_angles )
```

It sets the given input to output_joint_coordinates.

Parameters

_output_joint_angles	

Returns

None

3.1.2.8 set_output_coordinates()

```
void Forward_Kinematics::set_output_coordinates (
            const std::vector< double > & _output_joint_coordinates )
```

It sets the output_coordinates(input) to the output_joint_coordinates.

Parameters

_output_joint_coordinates	

Returns

None

3.1.2.9 solve_FK()

```
void Forward_Kinematics::solve_FK (
            const std::vector< double > & _input_joint_angles )
```

this function will calculate the end effector position rom the given input_joint_angles.

Parameters

input_joint_angles	these are the input joint angles of the robotic manipulator

The documentation for this class was generated from the following files:

- include/Forward_kinematics.hpp
- app/Forward_kinematics.cpp

3.2 Inverse_Kinematics Class Reference

The following Class contains all the methods,attributes of Inverse Kinematics Class. It provide methods to solve the inverse kinematics of a robotic manipulator.

```
#include <Inverse_kinematics.hpp>
```

Public Member Functions

- void solve_IK (const std::vector< double > &, const std::vector< double > &)
- void set_input_coordinates (const std::vector< double > &)
- void set_output_coordinates (const std::vector< double > &)
- void set_output_angles (const std::vector< double > &)
- void set_input_angles (const std::vector< double > &)
- void set_current_pose (const std::vector< double > &)
- void set_dh_a (const std::vector< double > &)
- void set_dh_d (const std::vector< double > &)
- void set_dh_alpha (const std::vector< double > &)
- std::vector< double > get_input_coordinates ()
- std::vector< double > get_output_coordinates ()
- std::vector< double > get_input_angles ()

- std::vector< double > get_output_angles ()
- std::vector< double > get_current_pose ()
- std::vector< double > get_dh_a ()
- std::vector< double > get_dh_d ()
- std::vector< double > get_dh_alpha ()
- void reset_pose ()
- std::vector< double > convert_input_angles_to_rotation_matrix (const std::vector< double > &)

3.2.1 Detailed Description

The following Class contains all the methods,attributes of Inverse Kinematics Class. It provide methods to solve the inverse kinematics of a robotic manipulator.

3.2.2 Member Function Documentation

3.2.2.1 convert_input_angles_to_rotation_matrix()

```
std::vector< double > Inverse_Kinematics::convert_input_angles_to_rotation_matrix (
            const std::vector< double > & input_joint_angles )
```

3.2.2.2 get_current_pose()

```
std::vector< double > Inverse_Kinematics::get_current_pose ( )
```

3.2.2.3 get_dh_a()

```
std::vector< double > Inverse_Kinematics::get_dh_a ( )
```

3.2.2.4 get_dh_alpha()

```
std::vector< double > Inverse_Kinematics::get_dh_alpha ( )
```

3.2.2.5 get_dh_d()

```
std::vector< double > Inverse_Kinematics::get_dh_d ( )
```

3.2.2.6 get_input_angles()

```
std::vector< double > Inverse_Kinematics::get_input_angles ( )
```

3.2.2.7 get_input_coordinates()

```
std::vector< double > Inverse_Kinematics::get_input_coordinates ( )
```

3.2.2.8 get_output_angles()

```
std::vector< double > Inverse_Kinematics::get_output_angles ( )
```

3.2.2.9 get_output_coordinates()

```
std::vector< double > Inverse_Kinematics::get_output_coordinates ( )
```

3.2.2.10 reset_pose()

```
void Inverse_Kinematics::reset_pose ( )
```

3.2.2.11 set_current_pose()

```
void Inverse_Kinematics::set_current_pose (
            const std::vector< double > & _current_robot_pose )
```

3.2.2.12 set_dh_a()

```
void Inverse_Kinematics::set_dh_a (
            const std::vector< double > & _dh_a )
```

3.2.2.13 set_dh_alpha()

```
void Inverse_Kinematics::set_dh_alpha (
            const std::vector< double > & _dh_alpha )
```

3.2.2.14 set_dh_d()

```
void Inverse_Kinematics::set_dh_d (
            const std::vector< double > & _dh_d )
```

3.2.2.15 set_input_angles()

```
void Inverse_Kinematics::set_input_angles (
            const std::vector< double > & _input_joint_angles )
```

3.2.2.16 set_input_coordinates()

```
void Inverse_Kinematics::set_input_coordinates (
            const std::vector< double > & _input_joint_coordinates )
```

3.2.2.17 set_output_angles()

```
void Inverse_Kinematics::set_output_angles (
            const std::vector< double > & _output_joint_angles )
```

3.2.2.18 set_output_coordinates()

```
void Inverse_Kinematics::set_output_coordinates (
            const std::vector< double > & _output_joint_coordinates )
```

3.2.2.19 solve_IK()

```
void Inverse_Kinematics::solve_IK (
            const std::vector< double > & input_joint_coordinates,
            const std::vector< double > & input_joint_angles )
```

The documentation for this class was generated from the following files:

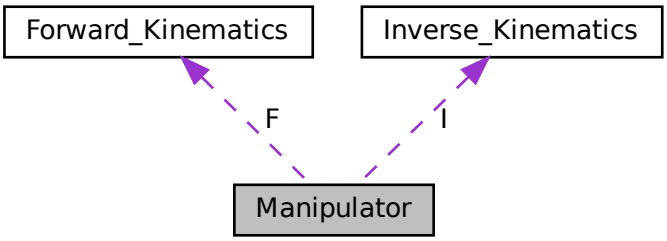
- include/Inverse_kinematics.hpp
- app/Inverse_kinematics.cpp

3.3 Manipulator Class Reference

This Class will call the Forward and Inverse Kinematics.

`#include <Manipulator.hpp>`

Collaboration diagram for Manipulator:



Public Member Functions

- void print_IK_solver ()

  It will print out the IK and FK for the given inputs.

Public Attributes

- Inverse_Kinematics I
- Forward_Kinematics F

3.3.1 Detailed Description

This Class will call the Forward and Inverse Kinematics.

3.3.2 Member Function Documentation

3.3.2.1 print_IK_solver()

```
void Manipulator::print_IK_solver ( )
```

It will print out the IK and FK for the given inputs.

Returns

None

3.3.3 Member Data Documentation

3.3.3.1 F

```
Forward_Kinematics Manipulator::F
```

3.3.3.2 I

```
Inverse_Kinematics Manipulator::I
```

The documentation for this class was generated from the following files:

- include/Manipulator.hpp
- app/Manipulator.cpp

Chapter 4

File Documentation

4.1 app/CMakeLists.txt File Reference

Functions

- add_executable (shell-app main.cpp Manipulator.cpp Inverse_kinematics.cpp Forward_kinematics.cpp) find_package(PythonLibs 2.7) target_include_directories(shell-app PRIVATE $
- target_link_libraries (shell-app ${PYTHON_LIBRARIES}) include_directories($

4.1.1 Function Documentation

4.1.1.1 add_executable()

```
add_executable (
            shell-app main.cpp Manipulator.cpp Inverse_kinematics.cpp Forward_kinematics.
cpp )
```

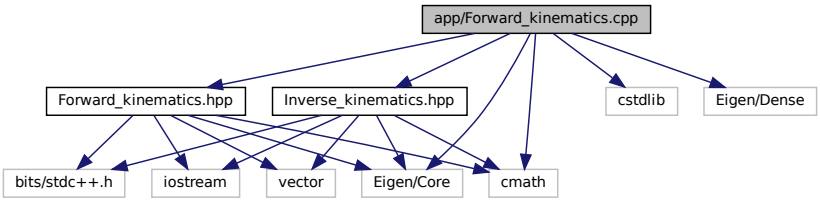
4.1.1.2 target_link_libraries()

```
target_link_libraries (
            shell-app ${PYTHON_LIBRARIES}  )
```

4.2 app/Forward_kinematics.cpp File Reference


```
#include "Forward_kinematics.hpp"
#include <cstdlib>
#include <cmath>
#include "Eigen/Core"
#include "Eigen/Dense"
#include "Inverse_kinematics.hpp"
```

Include dependency graph for Forward_kinematics.cpp:



4.2.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

BSD 3-Clause License

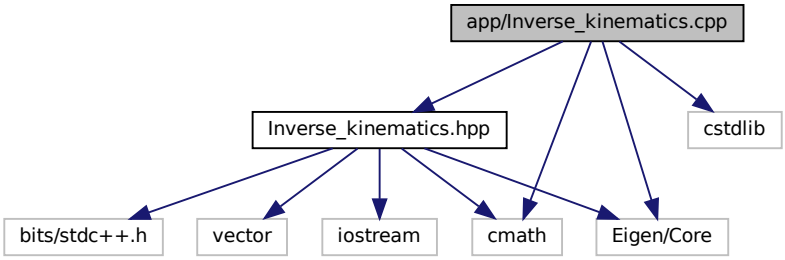
This file contains the Forward Kinematics methods used to find out the end-effector coordinates of the robotic manipulator.

4.3 app/Inverse_kinematics.cpp File Reference

BSD 3-Clause License Copyright (c) 2021, ACME Robotics, Rahul Karanam , Ameya Konkar All rights reserved.

```
#include "Inverse_kinematics.hpp"
#include <cstdlib>
#include <cmath>
#include "Eigen/Core"
```

Include dependency graph for Inverse_kinematics.cpp:



4.3.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

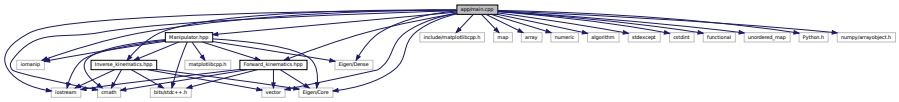
BSD 3-Clause License

This file contains the Forward Kinematics methods used to find out the end-effector coordinates of the robotic manipulator.

4.4 app/main.cpp File Reference


```
#include <iostream>
#include <iomanip>
#include <cmath>
#include "Eigen/Core"
#include "Eigen/Dense"
#include "Manipulator.hpp"
#include "Inverse_kinematics.hpp"
#include "Forward_kinematics.hpp"
#include "include/matplotlibcpp.h"
```

Include dependency graph for main.cpp:



Macros

- #define PI 3.14

Functions

- int main ()

  We use this main function to output the output joint coordinates for the given input_coordinates.

4.4.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

BSD 3-Clause License

This is our main source code file. It calls inverse Kinematics to implement our IK solver to simulate our path.

4.4.2 Macro Definition Documentation

4.4.2.1 PI

```
#define PI 3.14
```

4.4.3 Function Documentation

4.4.3.1 main()

```
int main ( )
```

We use this main function to output the output joint coordinates for the given input_coordinates.

Returns

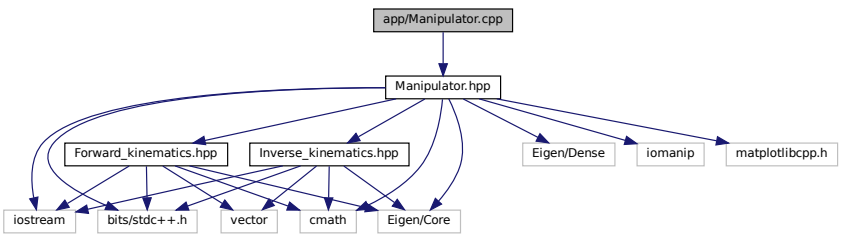
0;

4.5 app/Manipulator.cpp File Reference

BSD 3-Clause License Copyright (c) 2021, ACME Robotics, Rahul Karanam , Ameya Konkar All rights reserved.

```
#include "Manipulator.hpp"
```

Include dependency graph for Manipulator.cpp:



4.5.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

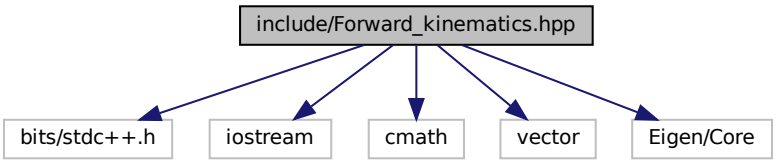
BSD 3-Clause License

This file contains the Forward Kinematics methods used to find out the end-effector coordinates of the robotic manipulator.

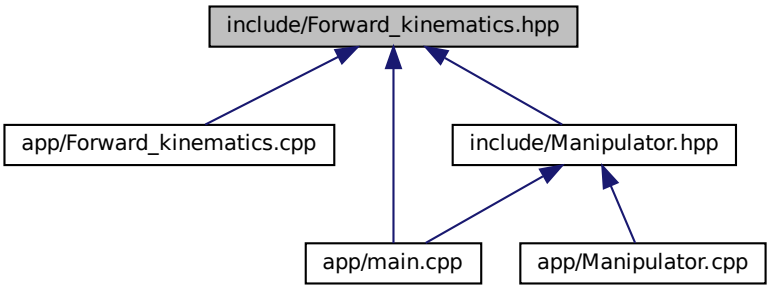
4.6 include/Forward_kinematics.hpp File Reference


```
#include <bits/stdc++.h>
#include <iostream>
#include <cmath>
#include <vector>
#include "Eigen/Core"
```

Include dependency graph for Forward_kinematics.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class Forward_Kinematics

  The following Class contains all the methods,attributes of Forward Kinematics Class. It provide methods to solve the forward kinematics of a robotic manipulator.

Macros

- #define PI 3.14

4.6.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

BSD 3-Clause License

This header file contains the Forward Kinematics class members and attributes Class to call solve_FK,getter and setter methods

4.6.2 Macro Definition Documentation

4.6.2.1 PI

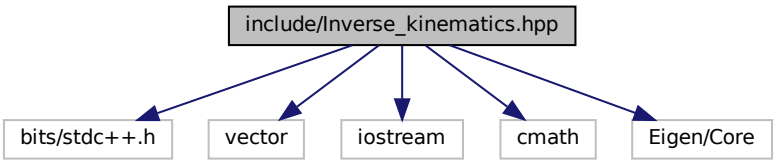
```
#define PI 3.14
```

4.7 include/Inverse_kinematics.hpp File Reference

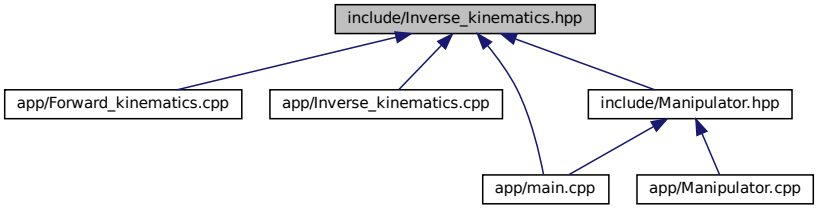
BSD 3-Clause License Copyright (c) 2021, ACME Robotics, Rahul Karanam , Ameya Konkar All rights reserved.

```
#include <bits/stdc++.h>
#include <vector>
#include <iostream>
#include <cmath>
#include "Eigen/Core"
```

Include dependency graph for Inverse_kinematics.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class Inverse_Kinematics

  The following Class contains all the methods,attributes of Inverse Kinematics Class. It provide methods to solve the inverse kinematics of a robotic manipulator.

Macros

- #define PI 3.14

4.7.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

BSD 3-Clause License

This header file contains the Forward Kinematics class members and attributes Class to call solve_FK,getter and setter methods

4.7.2 Macro Definition Documentation

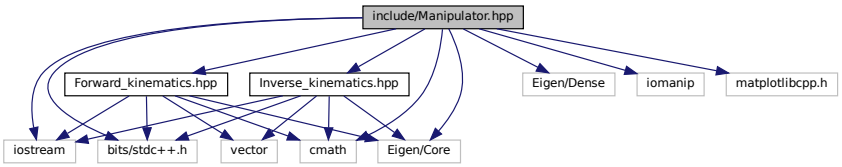
4.7.2.1 PI

```
#define PI 3.14
```

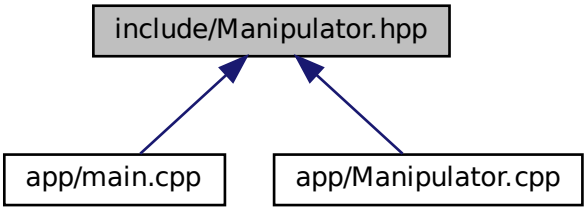
4.8 include/Manipulator.hpp File Reference


```
#include <bits/stdc++.h>
#include <iostream>
#include "Eigen/Core"
#include "Eigen/Dense"
#include <iomanip>
#include "Inverse_kinematics.hpp"
#include "Forward_kinematics.hpp"
#include "matplotlibcpp.h"
#include <cmath>
```

Include dependency graph for Manipulator.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class Manipulator

  This Class will call the Forward and Inverse Kinematics.

Macros

- #define PI 3.14

4.8.1 Detailed Description

Author

Rahul Karanam , Ameya Konkar

Copyright

BSD 3-Clause License

This Class will call the Forward Kinematics and Inverse Kinematics.

4.8.2 Macro Definition Documentation

4.8.2.1 PI

```
#define PI 3.14
```

Index