TUM Institute for Cognitive Systems (ICS)

OpenWalker

Module Description: Inverse Kinematics (IKM)

Florian Bergner, Emmanuel Dean, Rogelio Guadarrama-Olvera,
Simon Armleder, and Gordon Cheng

February 14, 2020

1 Module Description

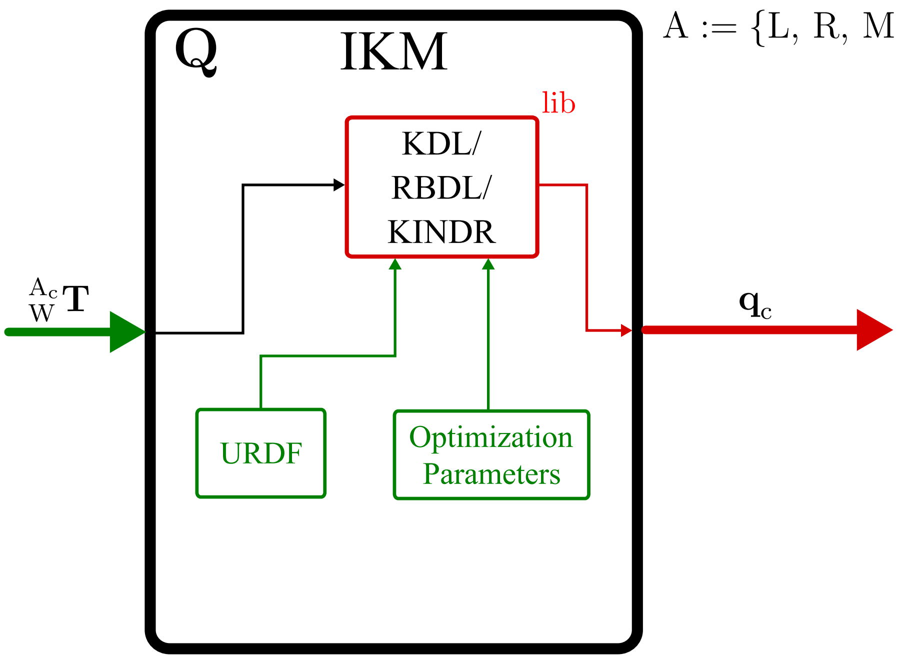

Figure 1.1: Inverse Kinematics module: This module implements the inverse kinematics for the robot.

The *Inverse Kinematics* module (IKM) computes the inverse kinematics of the robot. The inverse kinematics finds a set of joint positions such that a given set of robot end-effectors

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732287.

reach a specified Cartesian position. The OpenWalker framework uses one IKM to compute the commanded joint position $\mathbf{q}_\mathrm{c}$ such that the left and right foot, and the center-of-mass (CoM) of the real robot reach the desired Cartesian position ${}^{\mathrm{L_c}}_{\mathrm{W}}\mathbf{T}$, ${}^{\mathrm{R_c}}_{\mathrm{W}}\mathbf{T}$, and ${}^{\mathrm{M_c}}_{\mathrm{W}}\mathbf{T}$.

2 Module Connections

2.1 Inputs

Symbol	Name	Type	Description
${}^{\mathrm{L_c}}_{\mathrm{W}}\mathbf{T} \in \mathbb{R}^{4\times4}$	Left Commanded Foot Coordinate Frame	`HomogeneousTransformation`	This homogeneous transformation matrix transforms coordinates in the left commanded foot coordinate frame L to the world coordinate frame W.
${}^{\mathrm{R_c}}_{\mathrm{W}}\mathbf{T} \in \mathbb{R}^{4\times4}$	Right Commanded Foot Coordinate Frame	`HomogeneousTransformation`	This homogeneous transformation matrix transforms coordinates in the right commanded foot coordinate frame R to the world coordinate frame W.
${}^{\mathrm{M_c}}_{\mathrm{W}}\mathbf{T} \in \mathbb{R}^{4\times4}$	Commanded CoM Coordinate Frame	`HomogeneousTransformation`	This homogeneous transformation matrix transforms coordinates in the commanded CoM coordinate frame to the world coordinate frame W.

2.2 Outputs

Symbol	Name	Type	Description
$\mathbf{q}_\mathrm{c} \in \mathbb{R}^{DOF}$	Commanded Joint Position	`JointPosition`	This vector contains the next commanded joint positions for all the joints of the robot. The OpenWalker framework uses this module input to send position commands to the position controlled real robot.

2.3 Inter-Connections

The inputs of the IKM are connected to the outputs of the Command Generation Module (CmdGenM) which fuses the reference and offset coordinate frames of the left and right foot and the CoM to commanded coordinate frames. The output of the IKM is connected to the Real Robot Module (RRM) which sends the commanded joint positions to the position controlled real robot.

2.4 Common Methods

Similar to the Forward Kinematics Module (FKM), this module uses kinematic parameters such as joint properties (location, type), and link properties (location, length) to build up a rigid multi body system (MBS) that represents the kinematic model of the robot. Using this kinematic model, we can iteratively compute the inverse kinematics using a damped Levenberg-Marquardt method, also known as Damped Least Squares method. Therefore we repeatedly compute

$$\mathbf{q}_{\mathrm{c},k} = \mathbf{q}_{\mathrm{c},k-1} + \Delta\boldsymbol{\theta} \tag{2.1}$$

$$\Delta\boldsymbol{\theta} = \mathbf{J}^\top \left(\mathbf{J}\,\mathbf{J}^\top + \lambda^2\,\mathbf{I}\right)^{-1} \mathbf{e} \tag{2.2}$$

where

$$\mathbf{J}^{\#\lambda}(\mathbf{q}) = \mathbf{J}^\top \left(\mathbf{J}\,\mathbf{J}^\top + \lambda^2\,\mathbf{I}\right)^{-1} \tag{2.3}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732287.

is the damped pseudo inverse, $\mathbf{J}(\mathbf{q})$ the Jacobian, and $\mathbf{e}$ the error between the actual Cartesian positions and the target Cartesian positions [1, 2]. The iteration either stops when the error between actual and target Cartesian position is below the tolerance or when the step width

$$||\Delta\boldsymbol{\theta}||_2 \leq \delta_{\text{step}} \tag{2.4}$$

is below the acceptable step tolerance. The parameter λ is the damping factor and has to be chosen carefully.

References

[1] RBDL: An efficient rigid-body dynamics library using recursive algorithms, `https://rbdl.bitbucket.io/de/d92/group__kinematics__group.html#gaa5eabd37ff8b0925d2ecbf49fee1a8a7`.

[2] Di Vito, D., Natale, C., and Antonelli, G. (2017). A comparison of damped least squares algorithms for inverse kinematics of robot manipulators. IFAC-PapersOnLine, 50(1), 6869-6874.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732287.