MODELING THE ICUB WITH A KINEMATIC AND DYNAMIC LIBRARY

Semester project intermediary presentation

Julia Jesse

November 2008
The iCub robot:
small sized humanoid robot
European project among several universities, which aims to study cognition

http://robotcub.org
Goal of the project: biped gait stability.

Need:

- Model of the robot
  - Denavit-Hartenberg’s parameters
- Description of the movements
  - Forward vs. Inverse kinematic
- KDL (Kinematic and Dynamic Library)
Modeling body joint connectivity graph body joint connectivity graph
FLOATING BASE

Floating base

6 DoF joint
GEOMETRIC MODEL

end-effector

base frame

$A_{01}$

$A_{12}$
Idea: most common joint types can be characterized by a line in space.

Need only four parameters to locate a line:

- $a_i$: link (body) length
- $\alpha_i$: link twist
- $d_i$: link offset
- $\theta_i$: joint angle

Usually for revolute joints $a_i$, $\alpha_i$, $d_i$ are constants, $\theta_i$ is the only variable.
**Main features:**

- a base coordinate frame
- n joint axes
- an end-effector frame, embedded in the final body
Rules to place the DH coordinate frames:

- align $z_0$ and $z_{n+1}$ with the z axis of the base and end-effector frame
- align $z_i$ with joint axis $i$ (revolute joint: rotation axis)
- axis $x_i$ intersects $z_i$ and $z_{i+1}$ and is perpendicular to them
- $y_i$ is derived from $x_i$ and $z_i$, right-handed coordinate frame
Denavit-Hartenberg Parameters

- $a_i$: link (body) length
- $\alpha_i$: link twist
- $d_i$: link offset
- $\theta_i$: joint angle
FORWARD VS. INVERSE KINEMATIC

- **Forward kinematic:**
  - Determine the position of the end-effector knowing the joint angles -> unique solution

- **Inverse kinematic:**
  - Determine the joint angles knowing the position of the end-effector -> multiple solutions.

- Used in this project
KDL

- KDL = Kinematic and Dynamic Library
- Part of the Open Robot Control Project (Orocos)
- Goal: develop a free software and framework for robot and machine control
- KDL: Modeling and computing of kinematic chains such as robots
KDL AND THE ICUB

- Model arms and legs with kinematic chains using DH parameters (http://eris.liralab.it/wiki/ICubForwardKinematics)

  - API: Chains, made of Segments which contain a Joint to define rotation axis and a coordinate Frame expressed with DH parameters

- Forward kinematic
  - API: recursive algorithm to calculate the position from joint space to cartesian space.
Inverse kinematic

API : iterative algorithm based on Newton-Raphson iterations.

Idea: begin with an estimate $a'$ of joint angles.

Calculate forward position $p'$ and compare with input position $p$.

If the difference is very small : stop. Else calculate new estimate and continue.
Some Results

- Comparing forward kinematic found by KDL with matrix multiplication of transformation matrices defined with DH parameters.

  ➡️ Same results.

- Applying forward, inverse, forward kinematic.

  ➡️ Get the same end-effector position.

  ➡️ Get different joint angles (multiplicity of solutions).
QUESTIONS ?