Two-arms dual position control

H.-L. Pham[1] B.V. Adorno[2] V. Perdereau[1] P. Fraisse[2]

[1]Institut des Systèmes Intelligents et de Robotique - UPMC

[2]Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier - UM2

June 3, 2010

ASSIST project

Two arms manipulator with a computer vision system $\longrightarrow$ assistant robot for quadriplegic people

- Consortium:
  LIRMM, ISIR, CEA-LIST, LAAS
  et Union Mutualiste Propara
- Collaboration between LIRMM and ISIR: planning and control of the two arms manipulation

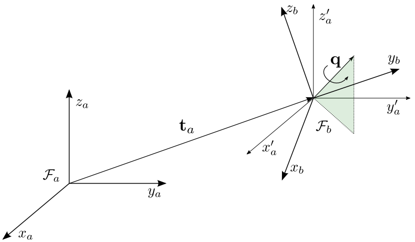
Outline

Context

- Different control modes:
  - Servo visual control for reaching the object
  - One arm control interacting with an object: position/force
  - Two arms coordination/manipulation: position/force
- Requirements:
  - Transparent transition between the modes
  - Singularity-free representation: NO Euler angles!

Rigid motions represented by dual quaternions

- Singularity-free
- Represent rotations and translations simultaneously
- "Simpler" than quaternions + Cartesian coordinates
- Use only 8 parameters
- Quaternions subjected to the Clifford Algebra ($\varepsilon \neq 0$, $\varepsilon^2 = 0$)
  - Dual quaternion multiplication: a term disappears!



$$\underline{\mathbf{q}} = \mathbf{q} + \varepsilon \frac{1}{2} \mathbf{t} \mathbf{q}$$

Rigid motions represented by dual quaternions

- "Commutativity" can be achieved by using the Hamilton operators:

$$\underline{\mathbf{q}}_a \underline{\mathbf{q}}_b \text{ equivalent to } \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_a)\underline{\vec{\mathbf{q}}}_b = \overset{-}{\mathbf{H}}(\underline{\mathbf{q}}_b)\underline{\vec{\mathbf{q}}}_a$$

- A sequence of rigid motions is given by a sequence of dual quaternion multiplications

Robot modeling in dual quaternion space

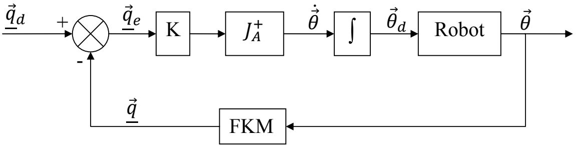
- Forward Kinematic Model:
  - Standard Denavit-Hartenberg applied in the dual quaternion space;

$$
\begin{aligned}
\underline{\mathbf{q}}_{\mathrm{DH}} &= \underline{\mathbf{q}}_{\mathrm{rot}}(z,\theta)\underline{\mathbf{q}}_{\mathrm{trans}}(z,d)\underline{\mathbf{q}}_{\mathrm{trans}}(x,a)\underline{\mathbf{q}}_{\mathrm{rot}}(x,\alpha) \\
\underline{\mathbf{q}}_{\mathrm{effector}} &= \underline{\mathbf{q}}_{\mathrm{DH}(1)}\underline{\mathbf{q}}_{\mathrm{DH}(2)}\cdots\underline{\mathbf{q}}_{\mathrm{DH}(n)}
\end{aligned}
$$

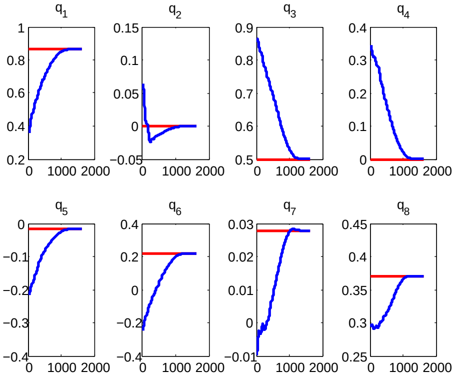
- Analytical Jacobian

$$
\frac{d\vec{\underline{\mathbf{q}}}}{dt} = \mathbf{J}_{\underline{\mathbf{q}}}\frac{d\vec{\theta}}{dt}
$$

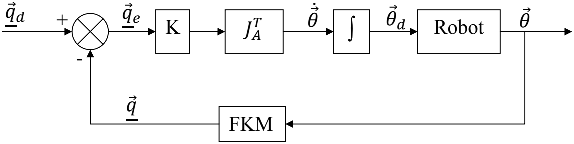
Dual position control: pseudo inverse Jacobian



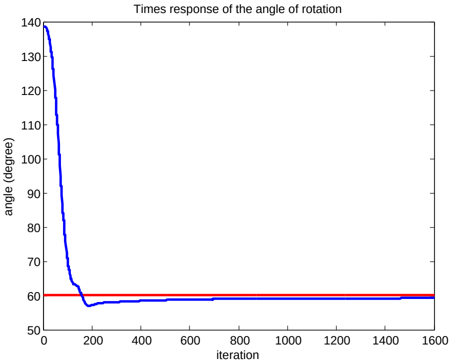
Asymptotically stable



Dual position control: transpose Jacobian



- Lyapunov: **Asymptotically stable**



Dual positions in the dual task-space

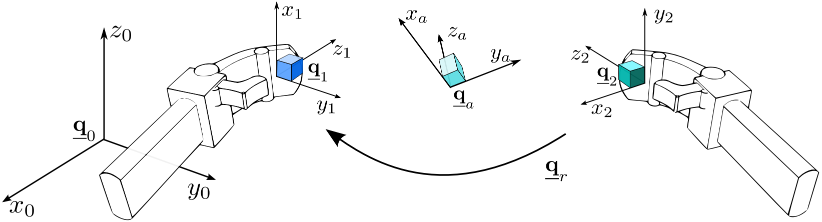
Definition

The relative and absolute dual positions can be defined as

$$\underline{\mathbf{q}}_r = \underline{\mathbf{q}}_2^* \underline{\mathbf{q}}_1$$

$$\underline{\mathbf{q}}_a = \underline{\mathbf{q}}_2 \underline{\mathbf{q}}_{\frac{r}{2}}$$

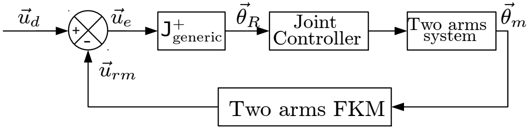
where $\underline{\mathbf{q}}_{\frac{r}{2}}$ corresponds to "half" of $\underline{\mathbf{q}}_r$

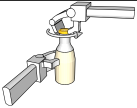
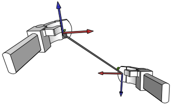
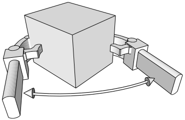
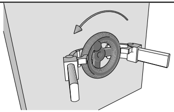


Control strategies for dual position control

- Basic idea:
  - Define the variables to be controlled;
  - Differentiate the equation;
  - Use the Hamilton operators to commute the terms;
  - Write the the input derivative as

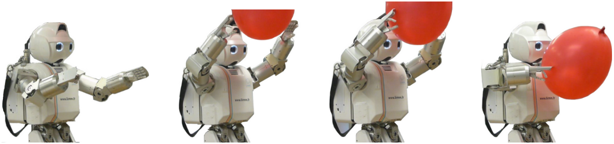
$$\dot{\vec{u}}_d = \mathbf{J}_{\text{generic}} \dot{\vec{\theta}}_R$$



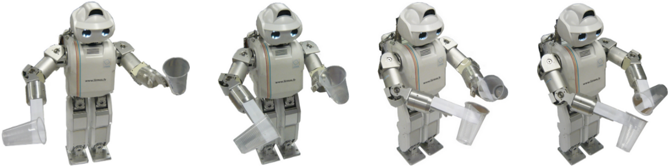
Strategy			
	$\vec{u}_{rd}$	$\mathbf{J}_{\text{generic}}$	Applicability
Relative dual position control	$\vec{\underline{\mathbf{q}}}_r$	$\mathbf{J}_{\underline{\mathbf{q}}_r} = \begin{bmatrix} \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_2^*)\mathbf{J}_1 & \overset{-}{\mathbf{H}}(\underline{\mathbf{q}}_1)\mathbf{J}_2^* \end{bmatrix}$	
Cartesian position control	$\vec{\mathbf{t}}_r$	$\mathbf{J}_{\text{cartesian}} = 2\overset{-}{\mathbf{H}}(\mathbf{q}_r^*)\mathbf{J}_{\mathbf{q}_r'} + 2\overset{+}{\mathbf{H}}(\mathbf{q}_r')\mathbf{J}_{\mathbf{q}_r}^*$	
Distance control	$\parallel \vec{\mathbf{t}}_r \parallel^2$	$\mathbf{J}_{\text{distance}} = 2\vec{\mathbf{t}}_r^T \mathbf{J}_{\text{cartesian}}$	
Full dual position control	$\begin{bmatrix} \vec{\underline{\mathbf{q}}}_a \\ \vec{\underline{\mathbf{q}}}_r \end{bmatrix}$	$\mathbf{J} = \begin{bmatrix} \mathbf{J}_{\underline{\mathbf{q}}_a} \\ \mathbf{J}_{\underline{\mathbf{q}}_r} \end{bmatrix}$	

Examples

- Grabbing a common object:



- Water pouring:



Ongoing works

- Servo visual control using dual quaternions: ISIR
- Forces in dual quaternion space: theoretical framework (LIRMM) and experiments (ISIR)
- Cooperative dual task-space: generalizations, redundancy, etc

Conclusions

- Dual position control using dual quaternions:
  - asymptotically stable
  - easy to implement
  - singularity-free representation
- Cooperative dual task-space
  - Based on dual quaternions
  - Meaningful variables defining the manipulation task
  - Control strategies sharing the same overall structure
  - Focus on the easiness of task definition