école normale supérieure paris–saclay

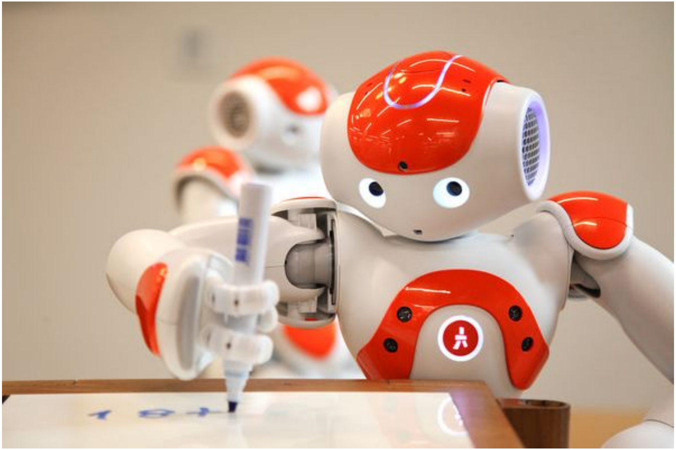
Writing with NAO

Adrien Bardes, Marius Dufraisse, Pierre Guetschel, Mengda Li

January 19, 2019

Goal of our project

We want to make our robot NAO write!



[1]

Methodology

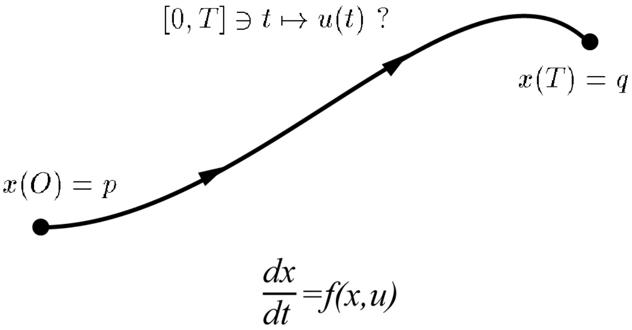
Analysis of handwriting and extraction of trajectory function

Inverse kinematics

- modeling the coordinate system
- finding the next “angles step” by computing jacobien

Analysis of handwriting and extraction of trajectory function

We formalize what we want to write by a trajectory function.

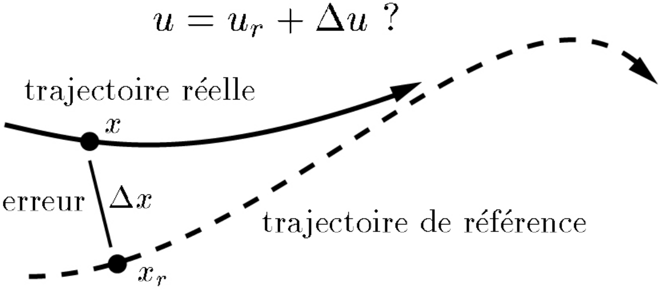


[2]

Inverse kinematics

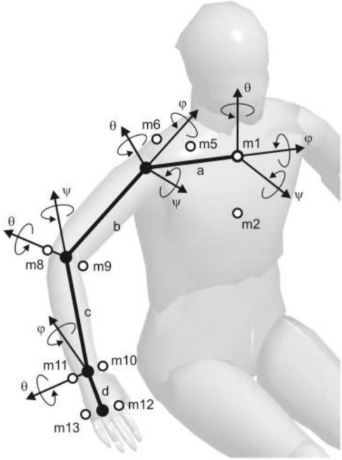
approching the goal trajectory

We approach this goal trajectory by solving a sequence of optimization problem: minimizing the errors betwenn the goal trajectory and the real trajectory.



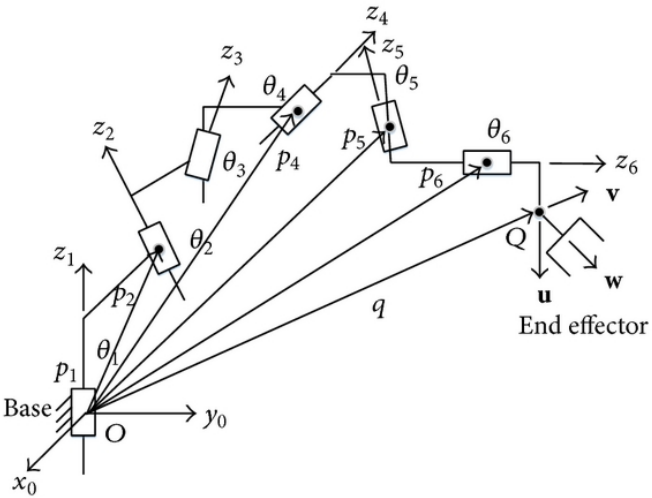
[2]

The robot is a n-joint system.
We find the position of end-effector (the pen) by composing a sequence of *change of coordinates* matrix.



[3]

modeling the coordinate system ii



[4]

finding the next "angles step" by computing jacobien i

Inverse Kinematics - Jacobian

$$\dot{Y} = \frac{\partial F}{\partial X} \dot{X}$$

$$V = J(\theta)\dot{\theta}$$

Desired motion of end effector

Unknown change in articulation variables

The *Jacobian* is the matrix relating the two: describing how each coordinate changes with respect to each joint angle in our system

[5]

finding the next “angles step” by computing jacobien ii

Inverse Kinematics - Jacobian

$$V = J(\theta)\dot{\theta}$$

$$V = \Big[\underbrace{v_x, v_y, v_z}_{\text{Change in position}}, \underbrace{\omega_x, \omega_y, \omega_z}_{\text{Change in orientation}}\Big]$$

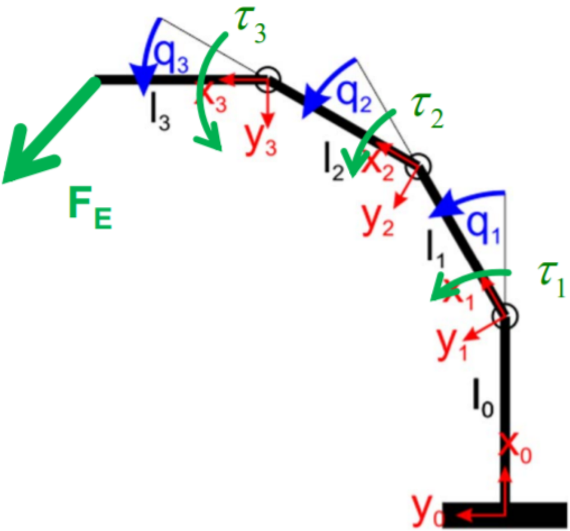
$$\dot{\theta} = \Big[\underbrace{\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6}_{\text{Change in articulation variables}}\Big]$$

Jacobian

$$J = \begin{bmatrix} \frac{\partial p_x}{\partial \theta_1} & \frac{\partial p_x}{\partial \theta_2} & \cdots & \frac{\partial p_x}{\partial \theta_6} \\ \frac{\partial p_y}{\partial \theta_1} & \cdots & & \\ \cdots & & & \\ \frac{\partial \alpha_z}{\partial \theta_1} & & & \frac{\partial \alpha_z}{\partial \theta_6} \end{bmatrix}$$

[5]

finding the next “angles step” by computing jacobien iii



[6]

finding the next "angles step" by computing jacobien iv

Algorithm 1 Numerical Inverse Kinematics

1: $\mathbf{q} \leftarrow \mathbf{q}^0$ ▷ Start configuration
2: **while** $\|\chi_e^* - \chi_e(\mathbf{q})\| > tol$ **do** ▷ While the solution is not reached
3: $\quad \mathbf{J}_{eA} \leftarrow \mathbf{J}_{eA}(\mathbf{q}) = \frac{\partial \chi_e}{\partial \mathbf{q}}(\mathbf{q})$ ▷ Evaluate Jacobian for $\mathbf{q}$
4: $\quad \mathbf{J}_{eA}^+ \leftarrow (\mathbf{J}_{eA})^+$ ▷ Calculate the pseudo inverse
5: $\quad \Delta\chi_e \leftarrow \chi_e^* - \chi_e(\mathbf{q})$ ▷ Find the end-effector configuration error vector
6: $\quad \mathbf{q} \leftarrow \mathbf{q} + \mathbf{J}_{eA}^+ \Delta\chi_e$ ▷ Update the generalized coordinates
7: **end while**

[6]

NAO robot illustrating a TechCrunch article.
https://www.robotlab.com/blog/nao-robot-illustrating-a-techcrunch-article

Planification et suivi de trajectoires.
http://cas.ensmp.fr/~petit/smai/

Interfacing of Kinect Motion Sensor and NAO Humanoid Robot for Imitation Learning.
https://www.youngscientistjournal.org/article/interfacing-of-kinect-motion-sensor-and-nao-humanoid-ro

Formal Kinematic Analysis of a General 6R Manipulator Using the Screw Theory

Bibliography ii

Matt Boggus. *Character Animation Forward and Inverse Kinematics*. https://slideplayer.com/slide/12902351/

Marco Hutter, Roland Siegwart, and Thomas Stastny. *Lecture «Robot Dynamics»: Summary*. https://www.ethz.ch/content/dam/ethz/special-interest/mavt/robotics-n-intelligent-systems/rsl-dam/documents/RobotDynamics2017/14-summary.pdf