UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI TECNICA E GESTIONE DEI SISTEMI INDUSTRIALI
CORSO DI LAUREA IN INGEGNERIA GESTIONALE

IMPERIAL COLLEGE LONDON

DEPARTEMENT OF MECHANICAL ENGINEERING
MECHATRONICS IN MEDICINE LAB

TESI DI LAUREA MAGISTRALE

KINEMATIC AND PERFORMANCE EVALUATION OF REDUNDANT MANIPULATORS

Relatore: Ch.mo Ing. GIOVANNI BOSCHETTI
Correlatore: Ch.mo Ing. RICCARDO SECOLI

Laureanda: LUCREZIA ANDRETTA

ANNO ACCADEMICO 2014-2015

“Ognuno è un genio.
Ma se si giudica un pesce dalla sua abilità
di arrampicarsi sugli alberi lui passerà tutta
la sua vita a credersi uno stupido”

Anonimo
(attribuito ad Albert Einstein)

Contents

SUMMARY

Lo scopo di questo elaborato è sviluppare un metodo per l'analisi delle performance cinematiche di robot seriali ridondanti.

Nello specifico si parlerà di ridondanza cinematica, cioè quando il manipolatore presenta più gradi di libertà rispetto al numero di variabili necessarie alla caratterizzazione di un determinato compito. In questo modo è possibile raggiungere un punto con più di una configurazione. Dunque, risulta interessante cercare di analizzare se tutte le configurazioni siano ugualmente valide o se ne esista una migliore.

Per fare ciò si ricorre alla creazione di un indice di performance, applicabile a un qualsiasi manipolatore seriale, anche ridondante, costruito sulla base della matrice Jacobiana. In particolare, l'indice massimizza la velocità cartesiana dell'organo terminale lungo una direzione, tenendo in considerazione la velocità dei singoli giunti.

Per questo lavoro, i risultati sperimentali sono stati verificati con il KUKA LWR 4+, un robot a sette gradi di libertà, disponibile al laboratorio di Meccatronica e Medicina dell'Imperial College di Londra.

The aim of this work is to propose a new method to analyse the kinematic performance of serial manipulators, even the redundant one.

A redundant robot has more degrees of freedom than the variables which define a specific task. Hence it is able to reach a point with more than just one arm configuration. So it is interesting to see if all the configuration are as good or if there is one better than the others.

To do so a performance index is created. It is based on the Jacobean matrix and aspire to maximize the Cartesian speed of the end effector fixing the joints' speeds.

The experimental investigation are carried out on the KUKA LWR 4+ robot. It is available at the Mechatronics in Medicine Lab at the Imperial College London.

INTRODUCTION

The serial robots are often used in the industrial sector, for tasks such as pick and place. Hence performance indexes are very important because they may provide useful hints in the design and optimization of the robots.

The redundant serial robot are particular because they are distinguished by more degrees of freedom than the degrees of freedom necessary to complete an assignments. Therefore, the performance indexes developed so far are not relevant. Indeed, these indexes just take a single configuration into account to optimize the direction of the task. While for redundant manipulator it is necessary to analyse the same direction changing the arm configuration.

In the first chapter the method to build the new index is explained. It is based on the Jacobean matrix and the linear programming to maximize the Cartesian speed of the end-effector, while the joints' speeds are known and fixed.

The index is implemented on the KUKA robot LWR 4+, which is introduced in the second chapter. While in the following chapter the kinematic analyses has been studied for the specific manipulator in order to build the robot's own index.

In chapter four the way the test have been carried out is specified both for numerical tests and experimental tests. Then the results are compared and the reliability of the index emerges.

1 DEVELOPMENT OF THE METHOD

The focus of this work is the kinematic performance analyses, so just kinematics indexes are taken into consideration. They are usually used to design robots, to optimize the execution of tasks and to compare robots of the same type.

1.1 TRADITIONAL PERFORMANCE INDEXES

Historically the indexes evaluate the motor's movement to shift the end-effector for a certain configuration.

The indexes for serial manipulators are generally based upon the Jacobean matrix. In particular, the first two indexes define the quality of the work envelope. Indeed, both indexes highlight problems near singularity configurations. They also provide global evaluation of robot performance mixing their translational and rotational capabilities. The third one provides uncoupled evaluations of robot translational capabilities along specific directions instead.

1.1.1 MANIPULABILITY

This index takes points of singularity into account. In these points the robot loses some or all the degrees of freedom, so it is not still able to carry out some tasks. However even near the singularity configurations the robot may not work right. Hence, the manipulator works better far away from these conditions, which are define by

$$\det(\boldsymbol{J}) = 0.$$

The Jacobean matrix is the way to measure the dexterity of a robot and the manipulability index, μ, can be define as

$$\mu=\sqrt{det(\boldsymbol{J}\boldsymbol{J}^T)}$$

According to Yoshikawa [1] it allows to find the best postures for manipulators. If μ has a value near zero it means that the robot reach an area where it may not work good, as it happens for the singularity configurations.

With this index it is possible to compare robots of the same type. Indeed, a good manipulator has big area of the workspace characterized with high values of the manipulability index.

1.1.2 CONDITION NUMBER

This index is based on the errors analysis. Specifically it expresses the transformation of a relative error from the joints' space to the Cartesian space:

$$k_J=\|\boldsymbol{J}\|\,\|\boldsymbol{J}-1\|=\frac{\sigma_{max}}{\sigma_{min}}$$

Where k_J is the condition number, σ_{max} and σ_{min} the maximum and the minimum singular value of the Jacobean matrix. More often, the inverse of the index is used

$$k_{rec}=\frac{1}{k_J}=\frac{\sigma_{min}}{\sigma_{max}}$$

In this way, it is possible to obtain values included between 0 and 1. When the value of k_{rec} is near 0 it means that the robot reaches a singularity configurations and the error is amplified to infinite; while near 1 input velocities and output velocities are the same.

1.1.3 DSI

The Direct Selective index allows evaluating independently the translational capabilities and the performances of the robot along the axis of its world reference frame [2]. So its formulation accounting for translations along a generic direction R, is

$$\mu_{\mathrm{R}} = \frac{1}{\sqrt{det(\boldsymbol{J}_{XR}^{-T}\boldsymbol{J}_{XR}^{-1})}}$$

$\boldsymbol{J}_{XR}^{-1}$ contains the velocity ratios between an end-effector translation along a generic direction R and the congruent joint rotations.

1.2 THE NEW PERFORMANCE INDEX

The new index is thought for any serial manipulator. Even the redundant ones. Redundancy imply an exaggerated abundance of something. In robot's field it is possible to talk about kinematic redundancy, which means that the dimensions of the operational space are minor than the dimensions of the joints' space. To simplify, the robot's degrees of freedom are greater than the variables necessary to define a specific task.

It is a relative concept because a robot can be redundant with respect to a task, but it can be not redundant for another assignment. However there are some redundant manipulator overall. They have one or more degrees of freedom than the six necessary to place and orient the end-effector in the space. Basically, they can reach a fixed point with more than just one arm configuration.

The redundancy involves some benefits:

- Increase dexterity and manipulability
- Avoid obstacles and kinematic singularities

- Minimize energy consumption
- Increase safety and reliability
- Optimize travel time and requested torque

Unfortunately there are some disadvantages:

- Greater structural complexity
- Problematic inverse kinematic and control algorithms

Therefore this kind of manipulators require an appropriate index to analyse their performance.

As usual the index is based on the computation of the Jacobean matrix.

1.2.1 VELOCITY JACOBEAN MATRIX

The velocities Jacobean matrix allows to relate the joints' speeds and the Cartesian speed of the end-effector and it depends on the robot's configuration.

There are two type of Jacobean matrix, the analytical Jacobean matrix and the geometrical one. The last one will be considered in this work. It is found calculating the velocity contributions of each joints with respect to the angular and linear speed of the end-effector.

The main use of the velocities Jacobean matrix is in the velocity kinematic problem, which can be evaluated as follows

$$\begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix} = \boldsymbol{J}\dot{\boldsymbol{q}}$$

This $\mathbf{J}$ matrix has m rows, which are the degrees of freedom in the Cartesian space, while the n columns are the manipulator's number of joints. Hence, the Jacobean matrix dimension is m x n. For redundant manipulator it is not a square matrix.

This equation can be used when the reference frame of the end-effector is the same as the global one. When the two coordinate systems are different there is the need to relate the two reference frames.

Let's consider a generic manipulator and highlight the two reference systems, the global one and the one relative to the end-effector.

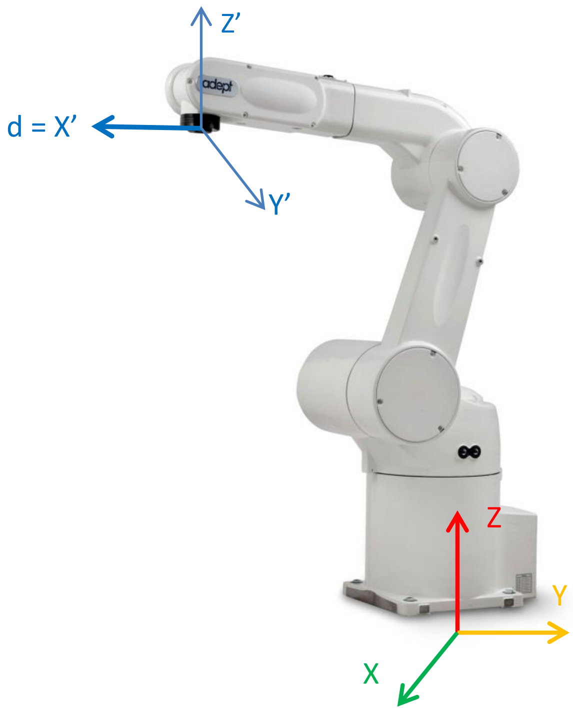

Fig.1.1: *Generic manipulator to implement the new index*

To evaluate the end-effector translation along the d direction, regarding the world reference frame, there is the need of a matrix. It is a rotational matrix $\mathbf{M}$, which defines the rotation of the x' axis than the x axis. In this way the new x-axis corresponds to the direction to evaluate. $\mathbf{M}$ is a 3 x 3 matrix and relates the two reference frames with the following expression

$$\Sigma' = \mathbf{M}\,\Sigma^{A}.$$

Where Σ' and Σ^{A} are respectively the relative reference frame of the end-effector and the word reference frame.

If the manipulator is moving on a plane, $\mathbf{M}$ is composed by one rotation. If the robot is moving in the space, $\mathbf{M}$ is composed by two rotations along independent axes.

Now that the position problem is solved, the velocity problem has to be evaluated. As said before, the aim is to find the Cartesian velocities knowing the joints' velocities. If the end-effector is moving just along the d direction, the problem is simplified because just the velocity along that direction has to be found, while all the others velocities are zero. As the reference frame of the end effector is rotated than the world reference frame there is the need of a rotational matrix, but in this case it is a 6 x 6 matrix build in a very easy way:

$$\mathrm{R} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{bmatrix}$$

The velocity problem can be express as

$$\begin{Bmatrix} \dot{x}_d \\ \mathbf{0} \end{Bmatrix} = \boldsymbol{R}\,\boldsymbol{J}\,\dot{\boldsymbol{q}}$$

Where the product ***R*J*** will be compacted in the following form

$$\mathrm{J_R} = [\mathrm{R}]\,[\mathrm{J}] = \begin{bmatrix} j_{R11} & \cdots & j_{R1n} \\ \vdots & \ddots & \vdots \\ j_{Rm1} & \cdots & j_{Rmn} \end{bmatrix}$$

With m the number of columns and n the number of rows.

Therefore emerge that the velocity along the d direction depends on the first row of $\mathbf{J}_{\mathrm{R}}$ matrix, calls $\mathbf{j}_{\mathrm{d}}$. So the velocity problem can be simplify

$$\max \dot{\mathrm{x}}_{\mathrm{d}} = \boldsymbol{j}_d^T\,\dot{\boldsymbol{q}}.$$

The aim of the new index is to estimate the maximum Cartesian velocity along the desired direction, optimizing the joints' velocity.

1.2.2 LINEAR PROGRAMMING

There are countless optimization methods in the literature. But the one chose in this work is the linear programming, because it is the most reliable and it has admissible computational times.

The linear programming allows to resolve optimization problems through a specific algorithm. It maximizes a function considering some constrains that limit the function.

In this case the problem, expresses in a linear programming way, is:

$$\text{maximize } \dot{x}_d = j_d\,\dot{q}$$

$$\text{s.t:}\quad \begin{cases} \{\mathbf{0} = \boldsymbol{J}_0\,\dot{\boldsymbol{q}} \\ \dot{q}_{min} \leq \dot{q}_i \leq \dot{q}_{max} \end{cases}$$

Where $\dot{q}_{\min}$ and $\dot{q}_{\max}$ are the lower and upper bounds of the joints' velocities; $\mathbf{j}_d$ the first row of the velocities matrix $\mathbf{J}_R$ and $\mathbf{J}_0$ all the other rows of the matrix.

Solving this problem allows to predict the robot's behavior in an easy way than the traditional performance indexes.

This method can be applied to any manipulators, even the redundant ones.

2 KUKA LWR 4+

Kuka is a German world's leading suppliers of robotics as well as plant and systems engineering. Kuka offers a wide range of industrial robots and robot system, among which we are interested in the lightweight robot. Kuka's work looks at a new generation of robots for "intelligent industrial work assistant". The robotic innovation with sensory capabilities for safety, fast teaching and simple operator control, opens up new areas of application in the vicinity of humans that were previously off-limits for robots [3]. This is possible thanks to their human-like behavior, which allow the human subject to be able to understand robot's intentions and collaborate with it.

The development of the lightweight robot has its roots in the 1993 ROTEX space shuttle mission, which demonstrated for the first time a robot arm in space that could work both by tele-operation from the ground and autonomously in space. To enable the astronauts to train for the mission they needed a comparable robot on Earth. The small lightweight robot was supposed to be based on the human model of an arm aiming at a weight-to-payload ratio of 1:1 and similar performance [4].

Indeed, the KUKA LWR (Light Weight Robot) is a particular robot which reproduce the human arm and which aim is to work closely with human operators thanks to its intuitive control, efficient programming and simple integration. It is the latest outcome of a bilateral research collaboration between KUKA Roboter GmbH and the Institute of Robotics and Mechatronics at the German Aerospace Center (DLR) [5]. It is use for cooperation, but it is not yet released for use in production, so that the purchasers largely come from the research sector and from advance engineering departments of companies which are looking to create new, more efficient production methods through the use of the LWR.

Figure 2.1 *Two Kuka robots simulating human arms*

It has seven revolute joints and this seven degrees of freedom with integrated sensors technology give it great flexibility, since they enable it to move around obstacles and reach even the most inaccessible places. The seven axes also help to avoid typical singularities of 6-axis kinematic system. For a given position and orientation, a pose can be selected that is favorable for the process concerned, as it is possible to see in the figure 2.

Figure 2.2 *KUKA LWR 7R*

The robot can be freely guided throughout its work envelope by hand thanks to its sensors and controls. Once reached the desire position , the robot maintains its pose. So it is simply to program it by teach. Its low weight and streamlined design make it flexible to use and ensure high cost-effectiveness [6].

The rounded design, which rules out any risk of crushing between structural components, contributes to the overall safety [4].

It is particularly suitable for complex assembly tasks and applications with direct human-machine cooperation. For example, at the Mechatronic in Medicine Lab they are studying how to use it during surgeries. It works very fine in orthopedic surgeries, the challenge is to use it in neurosurgeries. Basically the surgeon and the robot share the tool. Benefits are that the robot can be more accurate and can prevent human mistakes. In particular, the ACTIVE project, carried out by a consortium of fourteen institutes and universities, including Imperial College, London, aims to develop an integrated redundant robotic platform for neurosurgery that uses two cooperating robots which interact with the patient's brain [7].
Each joint is equipped with a position sensor on the input side and position and torque sensors on the output side, permitting position, velocity and torque control. Moreover, high stiffness is possible through active vibration damping. This aloud to the use of simple tools for any type of task [6].

Figure 2.3 *KUKA LWR can be used with several tools*

KUKA's principle components are: the base of the robot, the joint modules and the in-line wrist. Joint's modules consist of an aluminum structure that contain motors, gear units, brakes and sensors, as well as the necessary control and power electronics for seven axes. They also link the drives units to one other. While the in-line wrist is a two-axis wrist with two motors located in the last two axes, A5 and A6, as shown in figure 2.4. All the axes specifications are available in table 2.1

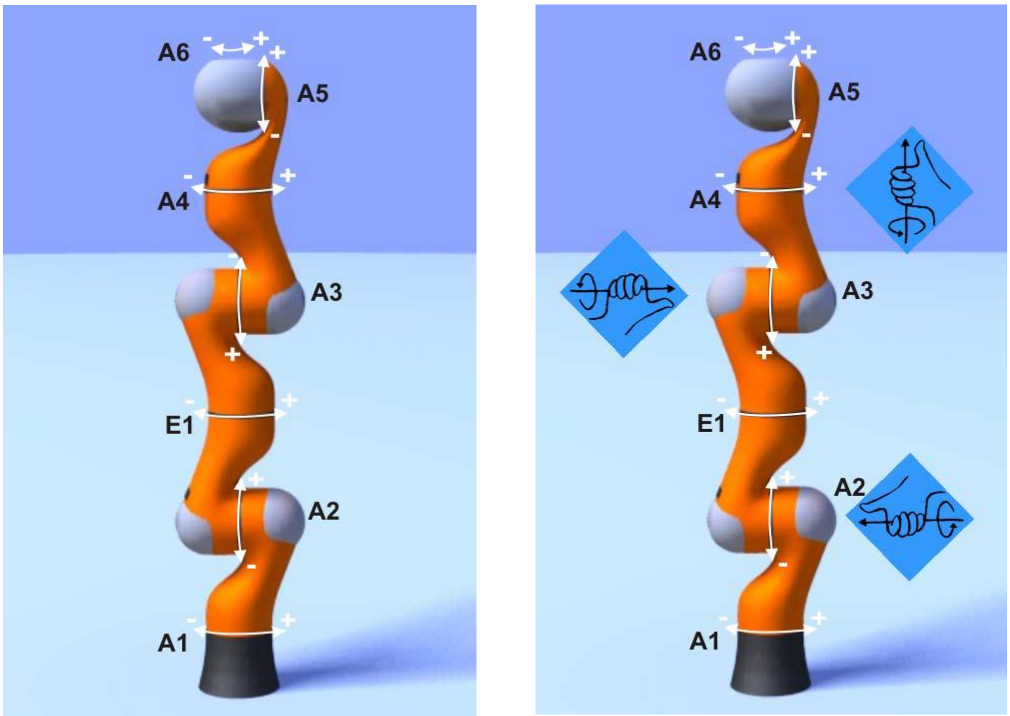

Figure 2.4: *KUKA stands in the home position for floor-mounted robot. Axes from A1 to A6 are the classic revolute joint even of anthropomorphic robots, while E1 is the redundant one. The direction of rotation of the axes is marked around the Z axis and it follows the right hand rule [8].*

AXES DATA	MOTION RANGE (deg)	SPEED WITH RATED PAYLOAD (deg/s)	MAXIMUM TORQUE (Nm)
A1	± 170°	112.5	200
A2	± 120°	112.5	200
E1	± 170°	112.5	100
A3	± 120°	112.5	100
A4	± 170°	180	100
A5	± 120°	112.5	30
A6	± 170°	112.5	30

Table 2.1 *KUKA axes specifications*

2.1 KSS (KUKA SYSTEM SOFTWARE)

2.1.1 KCP (KUKA Control Pannel) TEACH PENDANT

To handle the KRC (Kuka robot controller) the company develop an operator interface: the KUKA teach pendant, which provide the already established programming and operation environment and look & feel for the industrial user and at the same time enabled the access to the lightweight robot technology with its unique performance characteristics [4].

Figure 2.5 *KUKA and KCP*

The KCP has all the control and display functions required for operating and programming the industrial robot.

In figure 2.6 is possible to see the layout in detail.

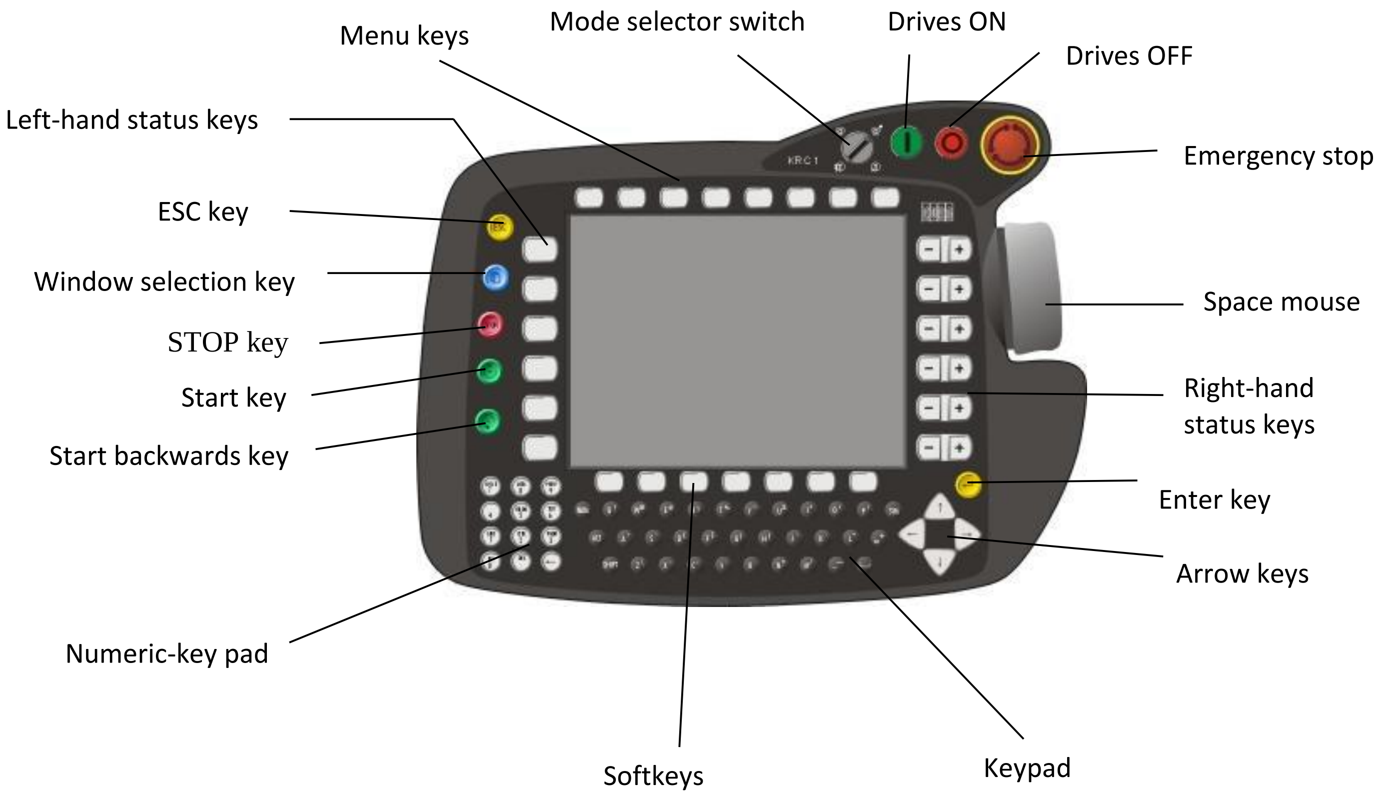

Figure 2.6 *Teach pendant layout*

Through the teach pendant the user can manipulate the robot and interface with the KUKA controller. The KCP is provided with Windows which supports the user interface of the KUKA System Software. It is called KUKA.HMI (KUKA Human-Machine Interface). The features are:

- User management
- Program editor
- KRL (KUKA Robot Language)
- Inline forms for programming
- Message display
- Configuration window
- Online help

2.1.2 MOTION PROGRAMMING

With the teach pendant it is easy to create a program and make the robot moves. The possible motion are point to point (PTP), linear (LIN) and circular (CIRC). To simplify the programming the inline forms are available for frequently used instructions. An example is shown in Figure 2.7.

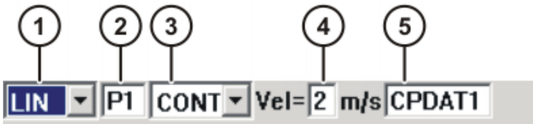

Figure 2.7 *Inline forms*

Where:

- 1 is the motion type (PTP, LIN, CIRC)
- 2 is the name of the end point, which is automatically generate and its coordinates are the coordinates of the end effector in that moment
- 3 indicates if the end point is approximated (CONT) or if the motion stops exactly at the end point (empty box)
- 4 is the velocity
- 5 is the name of the motion data set

Instructions can also be programmed without inline forms, using the KRL syntax.

For this work the programs have been created using the KRL syntax because in this way it was possible to decide the points with precision.

3 METHOD IMPLEMENTATION

To test the new implemented index some experiments have been carried out with the KUKA LWR robot.

Since the index is based on the kinematic of the robot, the KUKA LWR 4+ kinematic has been inspected.

3.1 KINEMATIC POSITION ANALYSIS

There are two type of kinematic position analysis: direct and inverse. In the first one the position of each joint is known and we want to know the end-effector position and orientation. To do this, a generic serial kinematic chain is analysed to know the end-effector position and orientation than a global frame. Indeed, an object position is identified by a relation between its reference frame and a global one fixes in the space.

The second type of kinematic position analysis works in the joint space. We give the robot the end-effector position and it elaborates the way to move the joints in the right way.

3.1.1 DIRECT KINEMATIC POSITION ANALYSIS

The first step in modeling robot is to determining DH parameters. DH parameters table is a notation developed by Denavit and Hartenberg, it is intended for the allocation of orthogonal coordinates for a pair of adjacent links in an open kinematic system. It is used in robotics, where a robot can be modeled as a number of related solids (segments) where the DH parameters are used to define the relationship between the two adjacent segments.

In this paper we refer to DH notation as the one modified by J.J.Craig and explains in his book [9]. The first step in determining the DH parameters is locating crank and determine the type of movement (rotation or translation) for each crank.

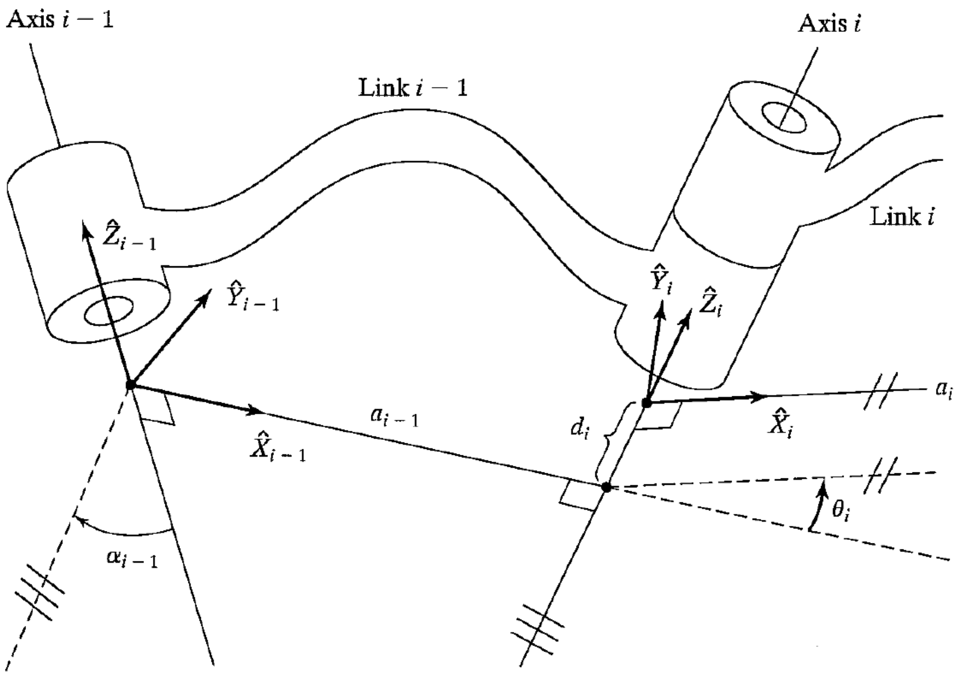

Figure 3.1 *Denavit-Hartenberg frame allocation*

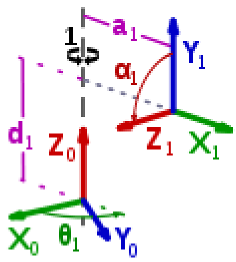

Figure 3.2 *D-H parameters*

The DH notation has two benefits. First, it is possible to allocate only one reference frame to each joint instead of two; second, the transformation matrix is just made by four elementary matrices not six.

To decide where to put the reference system of an axis the next axes must be taken into account, at the common normal and follow the next steps:

- the origin of the frame reference is unambiguously in the intersection between the joint axis and the common normal
- the z-axis is in the direction of the axis, the verse can be chosen
- the x-axis is in the direction of the common normal and it is oriented to the next axis
- the y-axis follows from the previous axes by choosing it to be a right-handed coordinate system

Once the references frames have been chosen, it is possible to obtain a matrix to move to the next axis with only four parameters. This parameters come from element movements:

- rotation around x-axis to have the two z-axes parallel: $\mathbf{R}_x(\alpha_{i-1})$
- translation along common normal to be in the next axis: $\mathbf{T}_x(a_{i-1})$
- rotation around z-axis to have the two x-axes parallel: $\mathbf{R}_z(\theta_i)$
- translation along z-axis to have the origins coincident: $\mathbf{T}_z(d_i)$

These are the four elementary matrices: two are always constant ($\mathbf{R}_x(\alpha_{i-1})$ and $\mathbf{T}_x(a_{i-1})$) and the other two may change, it depends on the type of joint. If the joint is revolute, $\mathbf{R}_z(\theta_i)$ changes and $\mathbf{T}_z(d_i)$ is constant; if the joint is prismatic vice versa. The changeable matrices are highlighted by a parameter.

Now it is possible to write the DH table with the four parameters. If they are constant there will be a number, otherwise there will be a parameter. The transformation matrix for each joint is the product of the four elementary matrices:

$$\mathbf{T}_{i,i-1} = \mathbf{R}_x(\alpha_{i-1}) * \mathbf{T}_x(a_{i-1}) * \mathbf{R}_z(\theta_i) * \mathbf{T}_z(d_i)$$

Otherwise there are some exceptions to handle and some particular cases to analyse. First exception is that the last axis doesn't have a next axis, so it is better to choose the simplest possible transformation matrix, which means a matrix with more possible 0. The z-axis has to be in the joint axis direction, but x-axis and y-axis can be anyone.

Also, it is better to choose the first frame than the global frame. In this way the two system will overlap when the parameter θ_1 is zero.

There are also some particular situations due to the relative position of the adjacent axes. Hence, if the two axes are parallel there are infinite common normal and DH is valid for each solution. Anyway there are better solutions, for example choose the common normal which passed through the robot link or, if the joints stand at different heights, choose always the same height.

If the two axes are accidents in a point the common normal degenerates in a point and x-axis must be chosen normal to both joint axes.

In the end, if the two axes are coincident it's better to choose a reference system to simplify the transformation matrix and x-axis must just be normal to z-axis.

With DH it is possible to typify completely a robot and draw the equivalent scheme of the cinematic chain.

For the KUKA LWR robot the DH parameters are shown in the table 3.1 below.

i	$T_{i,i-1}$	α_{i-1}	a_{i-1}	θ_{i-1}	d_{i-1}
1	$T_{1,0}$	0	0	θ_1	d_1
2	$T_{2,1}$	$\pi/2$	0	θ_2	0
3	$T_{3,2}$	$-\pi/2$	0	θ_3	d_3
4	$T_{4,3}$	$\pi/2$	0	θ_4	0
5	$T_{5,4}$	$-\pi/2$	0	θ_5	d_5
6	$T_{6,5}$	$-\pi/2$	0	θ_6	0
7	$T_{7,6}$	$-\pi/2$	0	θ_7	0

Table 3.1: *DH parameters for the KUKA LWR*

3.1.2 INVERSE KINEMATIC POSITION ANALYSIS

The KUKA LWR is a redundant robot, so theoretically there are infinite solutions for the inverse kinematic position problem. P. Artemiadis in his work [10] proposes a method to find just one solution. In order to define just one solution it is compare to a human arm. Therefore, the human arm swivel angle is used as a parameter for the robot arm to mimic the human behavior and the inverse kinematics is simplified.

Indeed, the human arm and the robot can be compared, like in the figure 3.3.

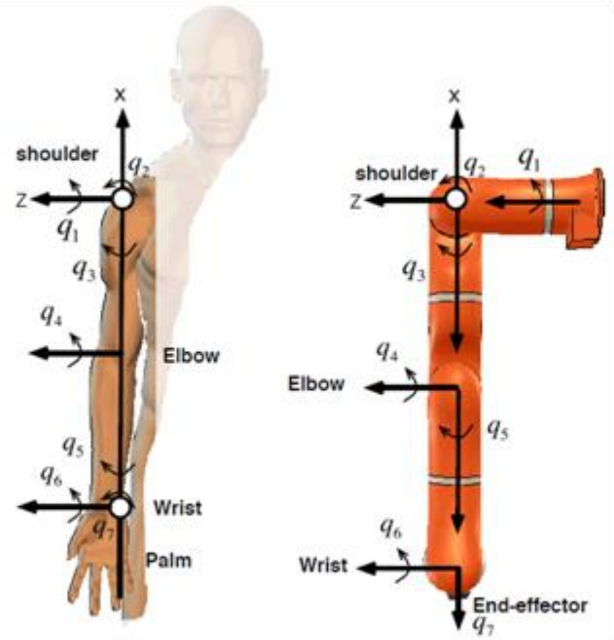

Figure3.3: *similarities between human arm and robot arm*

In both structures it is possible to identify shoulder, elbow and wrist joints connected by three rigid links.

It is possible to split the problem in two parts because the redundancy of the arm is actually found on the first four joints, that need to position the wrist. So we define a method to solve this four joints and the other three joints can be analytically solve as the spherical wrist of the anthropomorphous robot.

The aim of the inverse kinematic position problem is to calculate the joints' position (θ_i) giving the end-effector Cartesian position.

Therefore the problem resolution starts with knowing position and orientation of the end-effector with respect to the base frame. The position is defined by the coordinates (X_u, Y_u, Z_u), while the orientation by the rotation along the z-axis, y-axis and x-axis respectively (α_u, β_u, γ_u). Then the wrist position can be easily defined using a transformation matrix from the end-effector to the center of the wrist

$$\boldsymbol{T}_{U,7} = \boldsymbol{T}_z(d_7)$$

Where $\mathbf{T}_{U,7}$ is a 4×4 transformation matrix; $\mathbf{T}_z$ is a 4×4 translational matrix along the z-axis and d_7 is the length of the link.

But this is the wrist position in the end-effector system. To know the wrist position in the global frame another transformation matrix is required

$$\boldsymbol{T}_{W,0} = \boldsymbol{T}_U \, \boldsymbol{T}_{U,7}^{-1}$$

With $\mathbf{T}_U$ the transformation matrix from the end-effector reference frame to the world reference frame, which is known.

Now the position of the wrist is identified and its coordinates are (X_w, Y_w, Z_w).

The next step is to find the elbow position, which is the tough part because is the critical point due to the redundancy.

A simple physical interpretation of the redundant degree of freedom is based on the observation that, if the wrist is held fixed, the elbow is still free to swivel about a circular arc whose normal vector is parallel to the axis from the shoulder to the wrist. As the swivel angle ϕ changes, the elbow traces an arc of a circle lying on a plane which is perpendicular to the wrist-to-shoulder axis, as shown in figure 3.4.

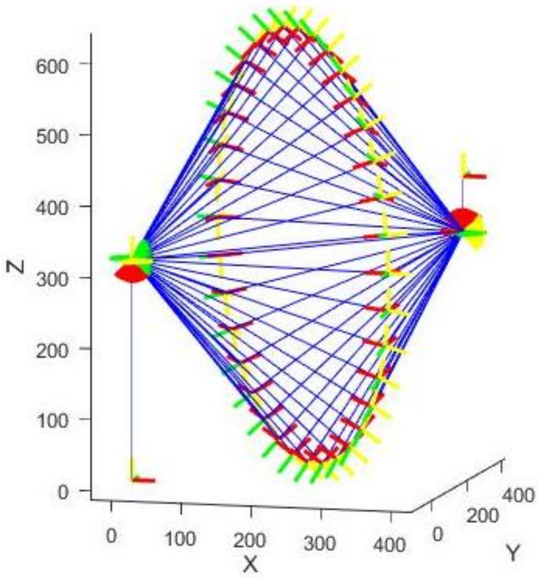

Figure 3.4: *arc traces by the elbow*

The swivel angle ϕ is an external parameter that the user can choose at the beginning with the coordinates of the end-effector.

Given ϕ, the center C of the circumference done by the elbow is easy to find geometrically because shoulder, elbow and wrist form a triangle shown in figure 3.5:

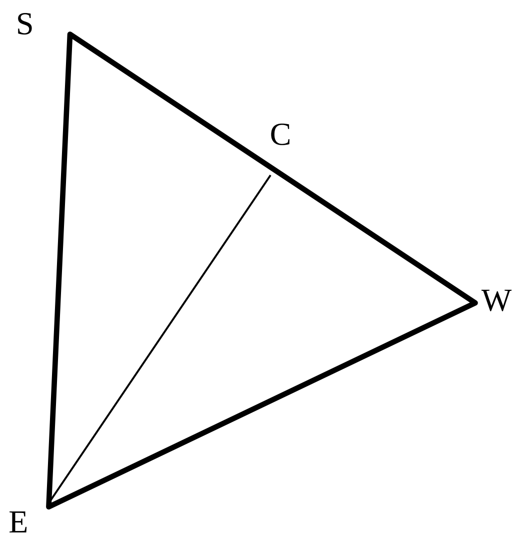

Fig.3.5: *simplified triangle made by the shoulder, elbow and wrist joints*

Hence, the geometrical rules are applied. The elbow-to-center axis is calculated as the height on the base, which is the shoulder-to-wrist axis:

$$CE = \frac{2\,A}{SW}$$

With A the area of the triangle, CE the elbow-to-center axis and SW the shoulder-to-wrist axis.

Then the shoulder-to-elbow projection on the base SW is identify with the Pythagoras theorem

$$SC = \sqrt{SE^2 - CE^2}$$

SC is the shoulder-to-center axis and SE is the shoulder-to-elbow axis.

As result, the coordinates of the center of the circumference can be estimated

- $X_c = \frac{X_W - X_S}{SW * SC}$
- $Y_c = \frac{Y_W - Y_S}{SW * SC}$
- $Z_c = \frac{Z_W - Z_S}{SW * SC}$

X_S, Y_S, Z_S are the coordinates of the shoulder. They are known because they are rigidly related to the world reference frame. There is only a translation along the z-axis as long as the link length.

To locate the elbow it is also necessary to define the vectors of the circumference. Some assumptions are taken: the perpendicular vector $\mathbf{n}$ to the circumference is defined as:

$$\boldsymbol{n} = [0, 0, 1]$$

and the directional vector $\mathbf{d}$ of the shoulder-to-wrist axis is identified by:

$$\boldsymbol{d} = \frac{[X_w - X_S,\ Y_w - Y_S,\ Z_W - Z_S\ ]}{SW}$$

Hence, it is possible to define the vectors of the circumference $\mathbf{u}$ and $\mathbf{v}$:

$$\boldsymbol{u} = \ -cross(n, d)$$

$$\boldsymbol{v} = \ cross(u, d)$$

Cross means that there is a cross product between the two elements.

Therefore, the position $\mathbf{P}_e$ of the elbow can be represented as:

$$\boldsymbol{P}_e = \ \boldsymbol{P}_c + \boldsymbol{v} * CE * \cos(\phi) + \boldsymbol{u} * CE * \sin(\phi)$$

$\mathbf{P_e}$ is the elbow position. It is the vector that contains (X_e, Y_e, Z_e) and $\mathbf{P_c}$ is the same for the center of the circumference.

With the positions of wrist and elbow, as well as the position and orientation of the end-effector, it is possible to analytically give a unique solution to the inverse kinematic problem and compute the seven joints' angles of robot.

The first two joints are easy to compute because they just depend on the elbow position:

$$\theta_1 = arctan2(X_e, Y_e)$$

$$\theta_2 = arctan2(Z_e, \sqrt{X_e^2 + Y_e^2})$$

The other joints are a bit more complicated to estimate, but just because some matrices have to be computed.

The third joint depends on its position in the world reference frame and the wrist coordinates position. Hence, the transformation matrix is calculated

$$\boldsymbol{T}_{3,0} = \boldsymbol{T}_{1,0} * \boldsymbol{T}_{2,1} * \boldsymbol{T}_{3,2}$$

where $\mathbf{T}_{1,0}$ and $\mathbf{T}_{2,1}$ are known because θ_1 and θ_2 have already been computed; $\mathbf{T}_{3,2}$ is not the DH transformation matrix, but it is simplified because only the rotation than the previous reference frame. So $\mathbf{T}_{3,2}$ is expressed with DH notation:

$$\boldsymbol{T}_{3,2} = DH(\pi/2\,, 0,0,0)$$

Then, the matrix can be computed as:

$$\boldsymbol{T} = \boldsymbol{T}_{3,0}^{-1} * \boldsymbol{P}_W$$

with $\mathbf{P}_W$ is the vector with the position of the wrist.

Now, it is possible to compute the rotation of the third joint:

$$\theta_3 = arctan2(\boldsymbol{T}(2), \boldsymbol{T}(1))$$

$\mathbf{T}(2)$ and $\mathbf{T}(1)$ are respectively the second and the first element of the **T** matrix computed previously.

The fourth joint is easy to estimate because the Carnot theorem is used and the rotation results as:

$$\theta_4 = \pi - arcos(\frac{EW^2 + SE^2 - SW^2}{2 * EW * SW})$$

For the fifth joint there is again the need to compute a transformation matrix. In this case it is:

$$\boldsymbol{B} = \boldsymbol{T}_{4,0}^{-1} * \boldsymbol{T}_U * \boldsymbol{T}_{U,7}^{-1}$$

Hence, the joint's variable is:

$$\theta_5 = arctan2(\boldsymbol{B}(3,3), -\boldsymbol{B}(1,3))$$

The last two joints have been computed in the same way. For the sixth joint the matrix and the joint's variable are:

$$\boldsymbol{C} = \boldsymbol{T}_{5,4}^{-1} * \boldsymbol{B}$$

$$\theta_6 = arctan2(\boldsymbol{C}(1,3), \boldsymbol{C}(3,3))$$

While, for the seventh joint they are:

$$\boldsymbol{D} = \boldsymbol{T}_{6,5}^{-1} * \boldsymbol{C}$$

$$\theta_7 = arctan2(-\boldsymbol{D}(3,1), \boldsymbol{D}(1,1))$$

3.2 KINEMATIC VELOCITY ANALYSIS

The kinematic velocity analysis describes the connection between the joints' velocity and the Cartesian velocity of the end-effector. Those informations are described in the velocities Jacobean matrix.

The problem can be expressed by the equation:

$$\begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ w_x \\ w_y \\ w_z \end{Bmatrix} = \boldsymbol{J}\dot{\boldsymbol{q}}$$

$\dot{x}, \dot{y}, \dot{z}$ are the linear velocities of the end-effector; whereas w_x, w_y, w_z are the actual angular velocities of the end-effector.

For the KUKA LWR the Jacobean matrix $\mathbf{J}$ is a 6 x 7 matrix, because the robot has seven joints.

Now it is possible to apply the linear programming and find the maximum velocity for the end-effector moving in a specific directions. As said before, if the direction is not coincident with the x-axis of the global reference frame the Jacobian matrix has to be pre-multiply for a suitable rotation matrix $\mathbf{R}$. Hence, just the first row of the new matrix is maximized and the rest of the matrix has to be zero.

4 PERFORMANCE EVALUATION

4.1 EXPERIMENTAL TESTS

The experimental tests were carried out with the KUKA LWR robot.

A path has been gone through five times for each possible arm configurations in order to obtain an average velocity value without the intrinsic repeatability error.

Not all the configurations were possible because of joints work envelope limits.

The data has been collected using the optical device Optotrak Certus device. It has an optical tracking technology and it can track and analyse kinetics and dynamic motion in real-time with this powerful research-grade motion capture system that features exceptional spatial and temporal accuracy [11].

Fig.4.1:*Optotrak monitoring system on the left and sensors on the right*

The sensors have been mounted on the end-effector and the monitoring system around 1.80 meters far in order to have the best condition to measure the KUKA movement.

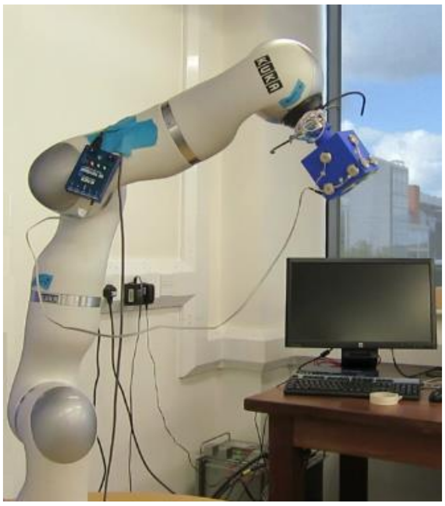

Fig.4.2: *KUKA robot with the sensors mounted*

Once the data have been collected, they have been handled with Matlab. Indeed the signal was affected by a lot of noises. Thanks to Matlab function filter it has been possible to obtain a more clear signal to use for the plots.

Actually, the Optotrak records just the sensors position. Knowing the sample time

$$s = \frac{1}{f}$$

With f the marker frequency of 1220 Hz, it is possible to estimate the sensors velocity. Since the velocity is the position derivative than the time.

4.2 NUMERICAL TESTS

The index algorithm was implemented with the Matlab software.

The kinematics of the specific robot was created and a punctual analysis in some points of the work envelope has been carried out. This points are the midpoints of the paths done with the KUKA LWR.

Also in this kind of tests just the possible configurations have been taken into account, as shown in figure 4.3.

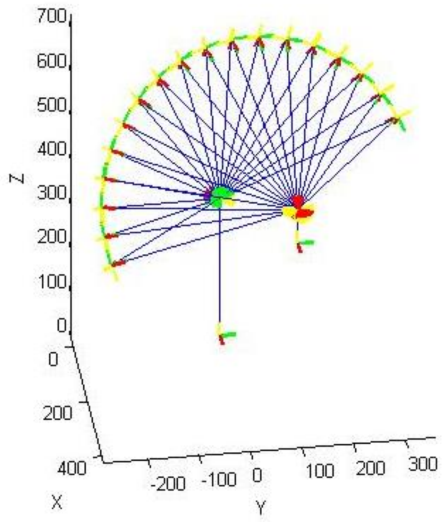

Fig.4.3: *Possible robot arm configuration portrayed with Matlab*

Thanks to the kinematic analysis the Jacobean matrix has been computed and using the Matlab function linprog the index has been implemented and plotted.

4.3 COMPARISON BETWEEN THEORETICAL AND REAL RESULTS

The diagrams with the two results for different paths are shown below.

The experimental results are highlight in red, while the theoretical ones are in blue.

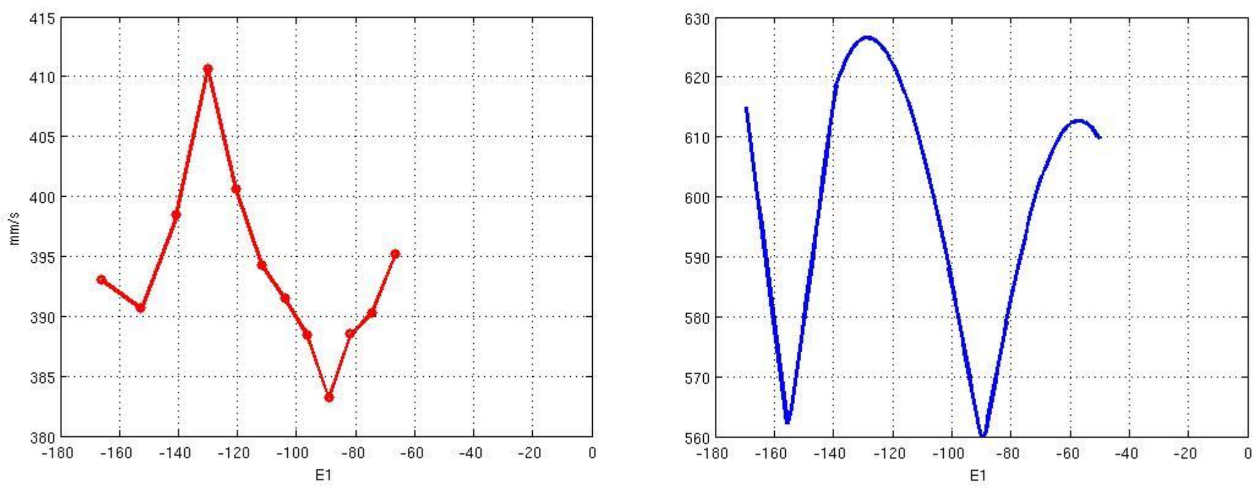

Fig.4.3: *Results for the path from A(216; 520; 400) to B(272; 576; 400) with right arm configurations*

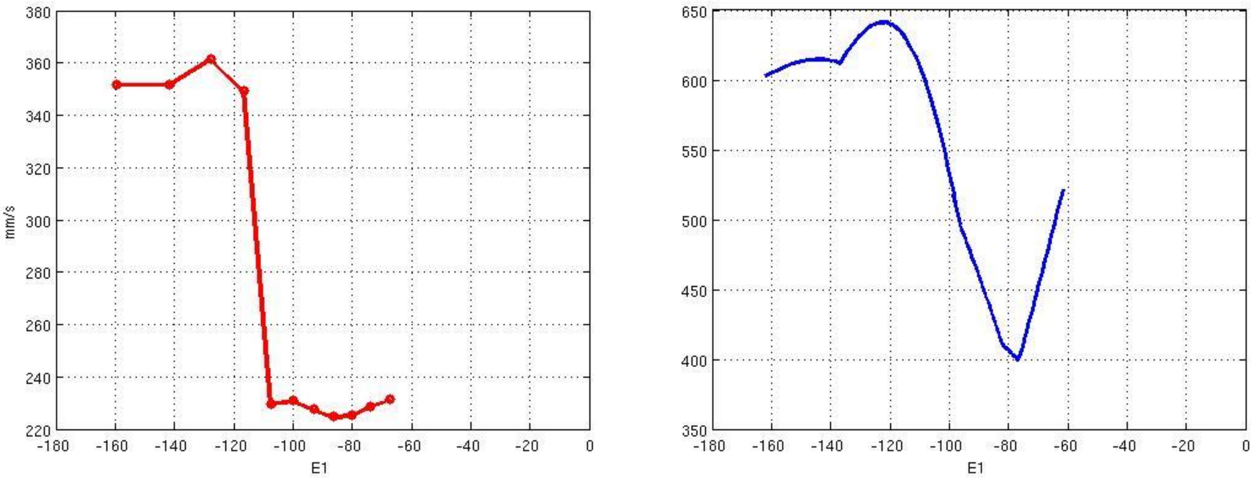

Fig.4.4: *Results for the path from A(157; 464; 500) to B(157; 524; 500) with right arm configuration*

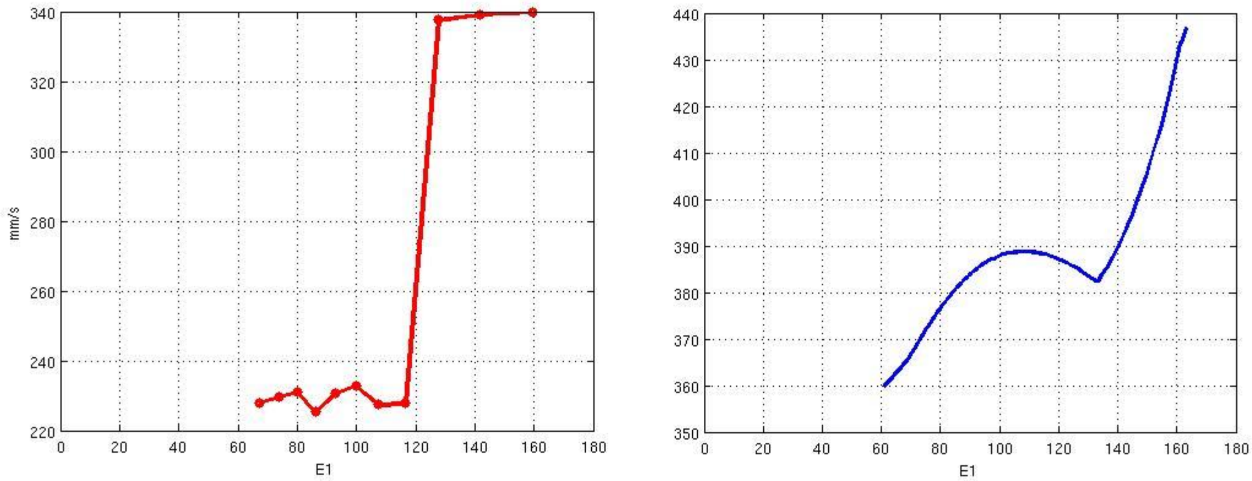

Fig.4.5: *Results for the path from A(157; 464; 500) to B(217; 464; 500) with left arm configuration*

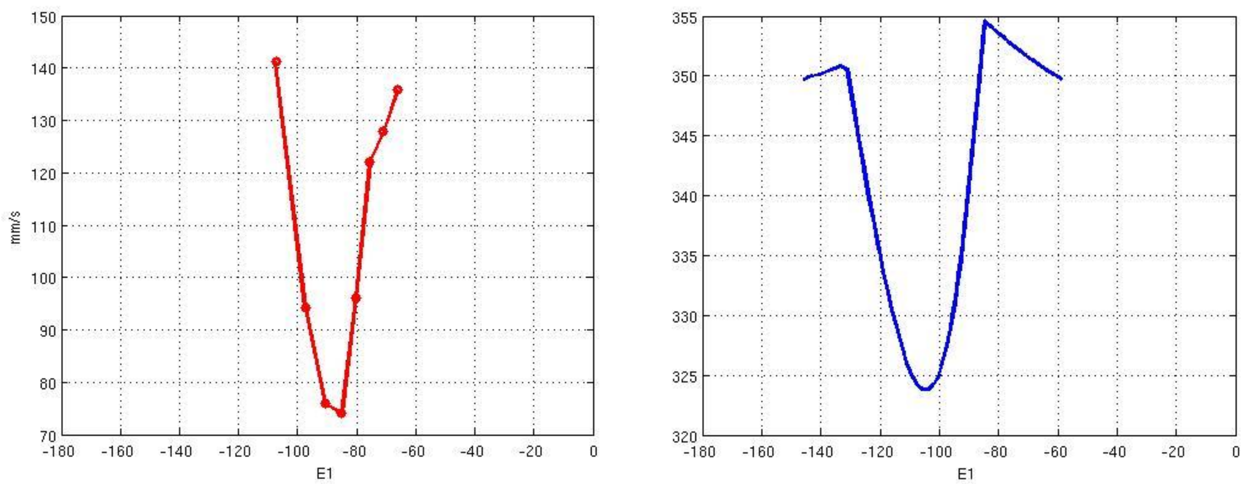

Fig.4.6: *Results for the path from a(393; 26; 600) to B(493; 26; 600) with right arm configuration*

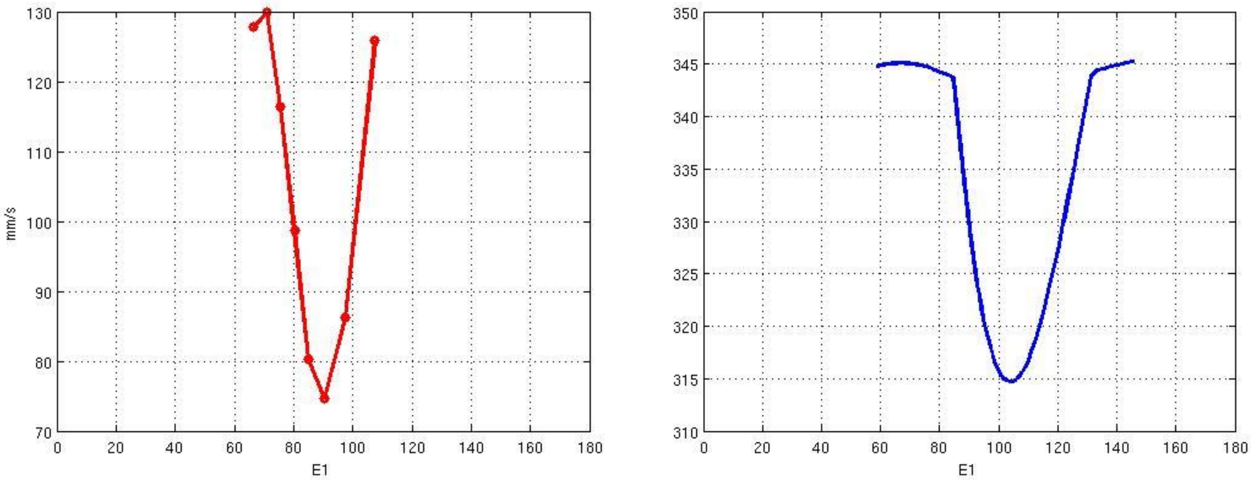

Fig.4.7: *Results for the path from A(393; 26; 600) to B(493; 26; 600) with left arm configuration*

It appears clear that the index is not perfect, but it gives useful informations about the robot's behavior.

As the first index used on redundant manipulators is quite a good start point for future analyses.

CONCLUSIONS

The performance indexes are useful for robot designers, manufactures and programmers.

In this paper a new purely kinematic performance index has been presented. It works for all the type of manipulator, but the aim here was to prove its reliability for redundant manipulators. This type of robot has more degrees of freedom than the Cartesian space, so the investigations were focused on maximizing the velocity along a specific direction changing the arm configuration.

The experimental investigations carried out on a KUKA LWR 4+ with seven revolute joints has proved that the proposed index formulation can provide reliable predictions of the robot performance. Overall the index gives useful information about the robot behavior, even though it is not perfect because of many approximation.

So far, the effectiveness of the index has been assessed on a single manipulator, but the index computation has been thought to have a general relevance. Indeed, just the Jacobean matrix has to be computed and the joints' velocities established

BIBLIOGRAPHY

[1] T. Yoshikawa, *"Manipulability of Robotic Mechanisms"*, The International Journal of Robotics Research 4, (1985) 3-9

[2] G. Boschetti, R. Rosa, A. Trevisani, *"Optimal robot positioning using task-dependent and direction-selective performance indexes: General definitions and application to a parallel robot"*, Robotics and Computer-Integrated Manufacturing 29 (2013) 431-443

[3] http://www.kuka-robotics.com/en/products/industrialrobots/sensitiv/lbr_iiwa_7_r800/start.htm

[4] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppe, A. Albu-Schäffer, A. Beyer, O. Eiberg, S. Haddadin, A. Stemmer, G. Grunwald, G. Hirringer *" The KUKA-DLR Lightweight Robot arm – a new reference platform for robotics research and manufacturing*", Robotics (ISR), 2010

[5] G. Schreiber, A. Stemmer, R. Bischoff *"The Fast Research Interface for the KUKA lightweight robot"*

[6] http://www.kukaconnect.com/wp-content/uploads/2012/07/KUKA_LBR4plus_ENLISCH.pdf

[7] http://www.active-fp7.eu/

[8] Kuka manual

[9] J. J. Craig "*Introduction to Robotics. Mechanics and Control"*, Pearson Education International, 2005

[10] Y. Wang, P. Artemiadis *"Closed-form inverse kinematic solution for anthropomorphic motion in redundant robot arms"*, Advances in Robotics & Automation, 2013

[11] http://www.ndigital.com/products/