Control of a Lower Mobility Dual Arm System

Philip Long, Wisama Khalil, Stéphane Caro

Control of a Lower Mobility Dual Arm System

P. Long W. Khalil S. Caro *

* *Institut de Recherche en Communications et Cybernétique de Nantes, UMR CNRS n° 6597 1 rue de la Noë, École Centrale de Nantes, 44321, Nantes, France. (e-mail: Wisama.Khalil@irccyn.ec-nantes.fr).*

Abstract: This paper studies the kinematic modeling and control of two cooperative manipulators. The cooperative system is composed of the two arms of the humanoid Nao robot of Aldebaran. The serial structure of each arm has five degrees of freedom, in the closed formulation, when transporting a common object, it has 4-DOF. The kinematic and dynamics representing the closed chain system is studied. A control scheme based on the cooperative task space with a minimum representation of the task is implemented. Furthermore by modeling the object grasp as a passive joint, all 6-DOF of the object can be controlled.

Keywords: Co-operative control, Robot control, Robotic manipulators

1. INTRODUCTION

The capability of dual arm system when processing parts reduces the need for custom fixtures and permits the use of a simpler end effector. The system can then execute sophisticated tasks that may be difficult for a single arm system. For example, rather than using a large serial robot a cooperative system distributes a heavy load among several smaller robots. Similarly if the object is of an unwieldy, non-rigid or awkward composition, the single arm robot may struggle to manipulate it.

By using a cooperative system, both the location and the internal forces of the object can be controlled. The two principal approaches to control such a system are hybrid position/force control and impedance control. An example of a position/force scheme for cooperative systems is found in Uchiyama and Dauchez (1988). The control variables are split into force and position controlled variables. The final control input is the sum of these variables after converting them to joint torque. Decentralized impedance control schemes, proposed by Bonitz and Hsia (1996) and Sadati and Ghaffarkhah (2008), control the internal forces for multi arm systems. An impedance relationship is enforced between each end effector velocity and the internal force at the grasp location. In Caccavale et al. (2008) impedance control schemes for cooperative manipulators are extended to manage the effect of the environmental forces on the object.

Another approach is to formulate kinematic relations that create a task space describing the multi-arm system while grasping an object. The main methods are known as *Symmetric formulation* Uchiyama and Dauchez (1988), and *cooperative task space* Chiacchio et al. (1996), Caccavale et al. (2000). The *cooperative task space* is a control scheme used to manage both the external motion of the object and the forces experienced by the object via an inverse kinematic controller. Furthermore it is demonstrated that an undesired motion in the object grasp can be represented as an extra joint in the kinematic chain, which increases the degree of redundancy or in this case increases the mobility of the object. On the other hand, the system can be viewed as a redundantly actuated parallel manipulator. Kinematic constraint equations are derived that establish a relationship between the chosen independent and dependent joint variables Yeo et al. (1999), Cheng et al. (2003). The dependent joint variables adopt values that ensure loop closure at each instant, the issue of inadmissible actuation schemes is discussed in Özkan and Özgören (2001).

Most of the preceding work has been carried out with dual arm systems, where both arms are either of planar or 6-DOF spatial structure. On the other hand the study of lower mobility cooperative manipulators has been limited. In Zielinski and Szynkiewicz (1996) admissible path planning for two 5-DOF robots is explored. In Bicchi et al. (1995) a generalized method based on the Jacobian matrix of each arm and their constraint relations with the object, permits the calculation of the mobility, possible first order differential motions, and manipulability velocity ellipsoids of general multiple limb robots. In Yeo et al. (1999) the cooperation between a 5-DOF and 4-DOF robot is used in conjunction with a passive joint in order to execute a 4-DOF position/force task. By adding a passive joint to the closed chain, the mobility is increased.

The main contribution of this paper is the formulation of two control approaches for lower mobility cooperative manipulators. Firstly by using the cooperative task space in conjunction with a minimum description of task Dombre (1981), the feasible directions of the object are controlled. The liberated DOF are then used to optimize extra criterion while satisfying the closed loop kinematic constraint equations. Secondly, by modeling the object grasp of each manipulator as a passive joint, the mobility of the object can be increased to 6-DOF while ensuring the loop closure. The cooperative control theory is applied to the two arms of Aldebaran NAO T14 humanoid robot, in simulation, and where possible experimentally. In this case, the object

has 4 DOF when firmly grasped by the manipulators. The system is modeled as a closed chain mechanism in Sections 2, 3, and 4, the control system is illustrated in Section 5.

2. DESCRIPTION OF THE SYSTEM

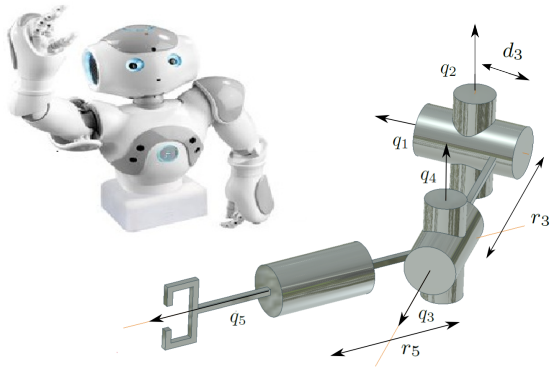

Fig. 1. NAO T14 (Courtesy of Aldeberan-Robotics) with schematic representation of its arm

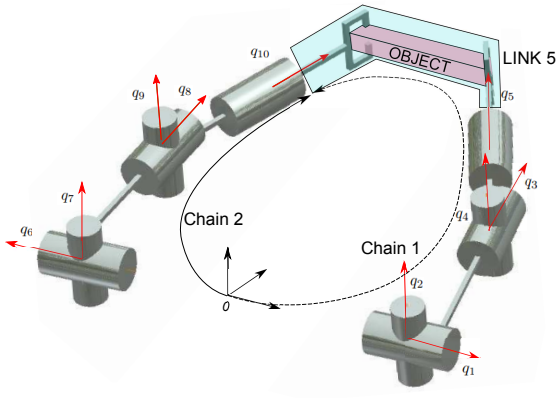

Fig. 2. Closed Loop Formulation

The NAO T14 robot illustrated in Fig.1 is the experimental platform used. The Modified Denavit-Hartenberg (MDH) notation from Khalil and Kleinfinger (1986) is used to describe the kinematics of system as given in Table 1. The right arm consists of joints 1-5 and the left arm of joints 6-10. Once the object is grasped a closed loop is formed from the two arms, the object, and the common robot torso. As illustrated in Fig. 2 link 5 of the closed chain is composed of link 5 and link 10 of the open chain and the object. Frame 10 is thus fixed on link 5. We introduce frame 11, that is equivalent to frame 10, but its antecedent is frame 5. The system has in this case only nine bodies. Hence joint 10 is denoted as the cut joint. The table parameters are explained as:

- j is the joint number, $a(j)$ is its antecedent joint
- σ_j is the joint type: revolute (σ_j=0), prismatic (σ_j=1) or fixed (σ_j=2)
- $\mathbf{z}_j$ is the jth joint axis
- $\mathbf{u}_j$ is the common normal between $\mathbf{z}_{a(j)}$ and $\mathbf{z}_j$
- $\mathbf{x}_j$ is perpendicular to $\mathbf{z}_j$ and one of the succeeding axes, $\mathbf{z}_k$ such that $a(k) = j$
- γ_j is the angle between $\mathbf{x}_{a(j)}$ and $\mathbf{u}_j$ about $\mathbf{z}_{a(j)}$
- b_j is the distance between $\mathbf{x}_{a(j)}$ and $\mathbf{u}_j$ along $\mathbf{z}_{a(j)}$
- d_j is the distance between $\mathbf{z}_{a(j)}$ and $\mathbf{z}_j$ along $\mathbf{u}_j$
- α_j is the angle between $\mathbf{z}_{a(j)}$ and $\mathbf{z}_j$ about $\mathbf{u}_j$
- θ_j is the angle between $\mathbf{u}_j$ and $\mathbf{x}_j$ about $\mathbf{z}_j$
- r_j is the distance between $\mathbf{u}_j$ and $\mathbf{x}_j$ along $\mathbf{z}_j$
- $\gamma_j = b_j = 0$ when $\mathbf{x}_{a(j)}$ is perpendicular to $\mathbf{z}_j$

Table 1. MDH Parameters of the closed loop

j	$a(j)$	$\sigma(j)$	$\gamma(j)$	b	d	α	θ	r
1	0	0	0	b_1	0	$-\frac{\pi}{2}$	θ_1	-r_1
2	1	0	0	0	0	$\frac{\pi}{2}$	θ_2	0
3	2	0	0	0	-d_3	$\frac{\pi}{2}$	θ_3	r_3
4	3	0	0	0	0	$-\frac{\pi}{2}$	θ_4	0
5	4	0	0	0	0	$\frac{\pi}{2}$	θ_5	r_5
6	0	0	0	b_1	0	$-\frac{\pi}{2}$	θ_6	r_1
7	6	0	0	0	0	$\frac{\pi}{2}$	θ_7	0
8	7	0	0	0	d_3	$\frac{\pi}{2}$	θ_8	r_3
9	8	0	0	0	0	$-\frac{\pi}{2}$	θ_9	0
10	9	0	0	0	0	$\frac{\pi}{2}$	θ_{10}	r_5
11	5	2	γ_{11}	b_{11}	d_{11}	α_{11}	θ_{11}	r_{11}

The parameters of frame 11 in Table 1 are defined by the grasp. Therefore, when the robot grasps the object, the transformation matrices of each serial arm are calculated. The parameters defining frame 11 with respect to frame 5 are calculated such that they yield ${}^{\mathbf{5}}\mathbf{T}_{\mathbf{11}}$.

$$ {}^{5}\mathbf{T}_{11} = \left({}^{0}\mathbf{T}_{5}\right)^{-1} {}^{0}\mathbf{T}_{10} \tag{1} $$

${}^{\mathbf{i}}\mathbf{T}_{\mathbf{j}}$ is the 4×4 transformation matrix from frame i to frame j. The six parameters defining frame 11 can be found by solving the following equation:

$$ \begin{aligned} {}^{5}\mathbf{T}_{11} = & \, rot_z(\gamma_{11}) \cdot trans_z(b_{11}) \cdot rot_x(\alpha_{11}) \cdot \\ & trans_x(d_{11}) \cdot rot_z(\theta_{11}) \cdot trans_z(r_{11}) \end{aligned} \tag{2} $$

where $rot_i(\theta)$ indicates a rotation of θ radians about the ith axis and $trans_i(l)$ a translation of l meters along the ith axis.

3. KINEMATIC MODELING

3.1 Kinematic Constraint Equations

The location, velocity, and acceleration of the frame at the cut joint must be equivalent when calculated via either chain. This ensures a constant object grasp throughout the trajectory. In the closed loop formulation, some joints may be actuated and some others could be passive. Let $\mathbf{q_a}$ contain the joint variables that are actuated, $\mathbf{q_p}$ contain the passive joint variables and $\mathbf{q_c}$ contain the passive joint variable where the chain is considered to be cut. The passive and cut joint variables can be obtained in terms of the active joint variables using the following geometric constraint equations:

$$ {}^{0}\mathbf{T}_{1}{}^{1}\mathbf{T}_{2}{}^{2}\mathbf{T}_{3}{}^{3}\mathbf{T}_{4}{}^{4}\mathbf{T}_{5}{}^{5}\mathbf{T}_{11} = {}^{0}\mathbf{T}_{6}{}^{6}\mathbf{T}_{7}{}^{7}\mathbf{T}_{8}{}^{8}\mathbf{T}_{9}{}^{9}\mathbf{T}_{10} \tag{3} $$

The kinematic constraints are given by:

$$\begin{bmatrix} \mathbf{v}_{11} \\ \omega_{11} \end{bmatrix} = {}^0\mathbf{J}_{11} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{bmatrix} = {}^0\mathbf{J}_{11}\ \dot{\mathbf{q}}_r \tag{4}$$

$$\begin{bmatrix} \mathbf{v}_{10} \\ \omega_{10} \end{bmatrix} = {}^0\mathbf{J}_{10} \begin{bmatrix} \dot{q}_6 \\ \dot{q}_7 \\ \dot{q}_8 \\ \dot{q}_9 \\ \dot{q}_{10} \end{bmatrix} = {}^0\mathbf{J}_{10}\ \dot{\mathbf{q}}_l \tag{5}$$

where $\dot{\mathbf{q}}_r$ and $\dot{\mathbf{q}}_l$ contain the joint velocities of the right arm and the left arm, respectively. ${}^0\mathbf{v_j}$ is the linear velocity and ${}^0\omega_j$ the angular velocity of frame j in frame 0, ${}^0\mathbf{J}_j$ is the kinematic Jacobian matrix of frame j w.r.t. frame 0. As frames 10 and 11 are coincident:

$$\begin{bmatrix} \mathbf{v}_{11} \\ \omega_{11} \end{bmatrix} = \begin{bmatrix} \mathbf{v}_{10} \\ \omega_{10} \end{bmatrix} \tag{6}$$

by substituting (4) and (5) into (6)

$$\begin{bmatrix} {}^0\mathbf{J}_{11} & -{}^0\mathbf{J}_{10} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} = 0 \tag{7}$$

By rearranging the rows and columns of (7), a relationship is obtained between the passive joint velocities and the actuated joint velocities

$$\begin{bmatrix} \mathbf{G}_a & \mathbf{G}_p & \mathbf{0} \\ \mathbf{G}_{ac} & \mathbf{G}_{pc} & \mathbf{G}_c \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_a \\ \dot{\mathbf{q}}_p \\ \dot{\mathbf{q}}_c \end{bmatrix} = 0 \tag{8}$$

that can be rewritten as

$$\mathbf{J}_c\, \dot{\mathbf{q}} = 0 \tag{9}$$

From the first row of (8), we obtain:

$$\dot{\mathbf{q}}_p = \mathbf{G}\ \dot{\mathbf{q}}_a \tag{10}$$

where

$$\mathbf{G} = -\mathbf{G}_p^{-1}\ \mathbf{G}_a \tag{11}$$

From the second row of (8), we obtain:

$$\dot{\mathbf{q}}_c = -\mathbf{G}_c^{-1}\,(\mathbf{G}_{ac} + \mathbf{G}_{pc}\ \mathbf{G})\,\dot{\mathbf{q}}_a \tag{12}$$

Upon differentiation of (8) with respect to time the acceleration constraints equation is expressed as:

$$\begin{bmatrix} \mathbf{G}_a & \mathbf{G}_p & \mathbf{0} \\ \mathbf{G}_{ac} & \mathbf{G}_{pc} & \mathbf{G}_c \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_a \\ \ddot{\mathbf{q}}_p \\ \ddot{\mathbf{q}}_c \end{bmatrix} + \dot{\mathbf{J}}_c\, \dot{\mathbf{q}} = 0 \tag{13}$$

3.2 Mobility Analysis

The mobility of a closed loop mechanism can be calculated exactly by using the rank of the closed loop kinematic constraint equations (9). The rank of $(\mathbf{J}_c)$ is obtained numerically using random configurations.

$$\mathbf{J}_c = \begin{bmatrix} {}^0\mathbf{J}_{11} & -{}^0\mathbf{J}_{10} \end{bmatrix} \tag{14}$$

The DOF can be found as the number of independent joints l before loop closure, minus those that lose their independence after the loop is closed (the degree of the constraint equation c):

$$n = l - c = l - rank(\mathbf{J}_c) \tag{15}$$

$$n = 10 - 6 = 4 \tag{16}$$

3.3 Closed Loop Jacobian Matrix

A new Jacobian matrix is derived that relates the actuated joint velocities to the task space velocities. The velocity is equal when calculated via either chain, from (4) and (5), and also via the actuators:

$$\begin{bmatrix} \mathbf{v} \\ \omega \end{bmatrix} = {}^0\mathbf{J}_{11}\ \dot{\mathbf{q}}_r = {}^0\mathbf{J}_{10}\ \dot{\mathbf{q}}_l = \mathbf{J}_a\ \dot{\mathbf{q}}_a \tag{17}$$

When a closed chain is formed, the object has 4-DOF. It can thus be fully controlled by four actuated joints, a feasible actuation scheme is chosen as $\mathbf{q}_a = [q_1\ q_2\ q_3\ q_7]^T$. The passive joint vector is $\mathbf{q}_p = [q_4\ q_5\ q_6\ q_8\ q_9\ q_{10}]^T$. Therefore using (10) and (12):

$$\dot{\mathbf{q}}_r = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \mathbf{G}(1,:) \\ \mathbf{G}(2,:) \end{bmatrix} \dot{\mathbf{q}}_a \tag{18}$$

where $\mathbf{G}(i,:)$ indicates the i^{th} row of the matrix $\mathbf{G}$. The Jacobian matrix $\mathbf{J}_a$ can be calculated by using the Jacobian matrix from either arm. For the right arm it is obtained by combining (17) and (18):

$$\mathbf{J}_a = {}^0\,\mathbf{J}_{11} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \mathbf{G}(1,:) \\ \mathbf{G}(2,:) \end{bmatrix} \tag{19}$$

$$\tag{20}$$

4. DYNAMIC MODELING OF CLOSED LOOP

4.1 Calculation of the Inverse Dynamic Model

The inverse dynamic model (IDM) calculates the active motor torques τ in terms of $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ of the joints. Let $\mathbf{q}_{tr} = \begin{bmatrix} \mathbf{q}_a^T & \mathbf{q}_p^T \end{bmatrix}^T$ denote the joint variables of the tree structure obtained when considering q_{10} as virtually cut. In order to find the Closed Loop Inverse Dynamic Model (CLIDM), first the IDM for the tree structure is found and then converted to CLIDM by using the following relation given in Khalil and Dombre (2004):

$$\tau = \left(\frac{\partial \mathbf{q}_{tr}}{\partial \mathbf{q}_a}\right)^T \mathbf{\Gamma}_{tr} = \mathbf{\Gamma}_a + \left(\frac{\partial \dot{\mathbf{q}}_p}{\partial \dot{\mathbf{q}}_a}\right)^T \mathbf{\Gamma}_p \tag{21}$$

where $\mathbf{\Gamma}_{tr}$ denotes the joint torques of the tree structure. It can be expressed as:

$$\mathbf{\Gamma}_{tr} = \begin{bmatrix} \mathbf{\Gamma}_a \\ \mathbf{\Gamma}_p \end{bmatrix} = \mathbf{A}_{tr}\,(\mathbf{q}_{tr}) \begin{bmatrix} \ddot{\mathbf{q}}_a \\ \ddot{\mathbf{q}}_p \end{bmatrix} + \mathbf{H}_{tr}\,(\mathbf{q}_{tr}, \dot{\mathbf{q}}_{tr}) \tag{22}$$

$\mathbf{\Gamma}_a$, $\mathbf{\Gamma}_p$ represent the torque on the actuated and passive joints of the tree structure, respectively. $\mathbf{A}_{tr}$ and $\mathbf{H}_{tr}$ are tree structure inertia matrix and the tree structure matrix of Coriolis, Centrifugal, and Gravity forces respectively. Using (10) and (21), we obtain the dynamic model:

$$\tau = \begin{bmatrix} \mathbf{I}_N & \mathbf{G}^T \end{bmatrix} \begin{bmatrix} \mathbf{\Gamma}_a \\ \mathbf{\Gamma}_p \end{bmatrix} \tag{23}$$

$\mathbf{G}$ is defined by (11) and $\mathbf{I}_N$ is the identity matrix of dimension N, where N is equal to the DOF of the system. Substituting the general expression for the tree dynamic

model given by (22) into (21), the closed loop dynamic model is obtained as:

$$\tau = \left[\mathbf{I}_N \ \mathbf{G}^T \right] \mathbf{A}_{tr} \begin{bmatrix} \ddot{\mathbf{q}}_a \\ \ddot{\mathbf{q}}_p \end{bmatrix} + \left[\mathbf{I}_N \ \mathbf{G}^T \right] \mathbf{H}_{tr} \tag{24}$$

4.2 Dynamic Model with Redundant Actuators

When the number of actuated joints is greater than four, the inverse dynamic model will be given by:

$$\tau_{ra} = \left(\frac{\partial \mathbf{q}_a}{\partial \mathbf{q}_{ra}} \right)^T \tau \tag{25}$$

where τ_{ra} is the torque vectors composed of *all* actuated joints. $\mathbf{q}_{ra}$ is defined as:

where $\mathbf{q}_r$ is the vector of redundant actuated joints. Thus, τ_{ra} can be obtained as:

$$\mathbf{q}_{ra} = \begin{bmatrix} \mathbf{q}_a \\ \mathbf{q}_r \end{bmatrix} \tag{26}$$

$$\tau_{ra} = \left[\mathbf{I}_N \ \left(\frac{\partial \mathbf{q}_r}{\partial \mathbf{q}_a} \right) \right]^+ \tau \tag{27}$$

4.3 Calculation of the Direct Dynamic Model

The direct dynamic model (DDM) calculates the independant joint accelerations $\ddot{\mathbf{q}}_a$ from the motor torques τ. It can be obtained after expressing $\ddot{\mathbf{q}}_p$ in terms of $\ddot{\mathbf{q}}_a$ as shown in (13), then substituting the result into (24).

5. CONTROL AND SIMULATION

5.1 Kinematic controller

A kinematic controller obtains the joint velocities that realize a desired task frame velocity. One implementation uses (17) to obtain the actuated joint velocities for a certain task velocity. Then from (8) the remaining velocities, which satisfy the closed loop constraints, can be resolved. Instead of this two step process, by using the *cooperative task space* as defined by Chiacchio et al. (1996), the task velocities and constraints can be embedded into one equation. This compact representation is applied for the first of our control schemes. Twelve variables are fully defined, six controlling the object location in space and a further six controlling the relative location between the end effectors, denoted here as the *task* and *relative* variables respectively. The direct kinematic model that relates the joint velocities to the end effector velocities is recalled here:

$$\begin{bmatrix} \mathbf{V}_{tsk} \\ \mathbf{V}_{rel} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{tsk} \\ \mathbf{J}_{rel} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} = \mathbf{J}_{ext} \begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} \tag{28}$$

where

$$\mathbf{V}_{tsk} = \begin{bmatrix} \mathbf{v}_{tsk} \\ \omega_{tsk} \end{bmatrix}, \ \mathbf{V}_{rel} = \begin{bmatrix} \mathbf{v}_{rel} \\ \omega_{rel} \end{bmatrix} \tag{29}$$

$$\mathbf{J}_{tsk} = \frac{1}{2} \left[{}^0\mathbf{J}_{11} \ {}^0\mathbf{J}_{10} \right] \ \mathbf{J}_{rel} = \left[-{}^0\mathbf{J}_{11} \ {}^0\mathbf{J}_{10} \right] \tag{30}$$

$$\mathbf{J}_{ext} = \begin{bmatrix} \mathbf{J}_{tsk} \\ \mathbf{J}_{rel} \end{bmatrix} \tag{31}$$

5.2 Kinematic Control with Minimum representation

Minimum representation is used when the location of the end effector does not need to be not fully defined, see Dombre (1981). The degree of redundancy is exploited to improve the performance of the robot with regard to some optimization criteria. Since the mobility of the system is 4, the maximum dimension of the task coordinates is 4, $dim(\mathbf{V}_{tsk}) \leq 4$. Therefore the task is defined as transporting a common object from one position to another meaning some variables do not need to be assigned. This leads to a minimum representation of the *task* variables denoted as $\dot{\mathbf{h}}$ while fully controlling all the *relative* variables. The minimum representation is implemented by pre-multiplying the Jacobian matrix of equation (31) by a selection matrix $\mathbf{H}$:

$$\begin{bmatrix} \dot{\mathbf{h}} \\ \mathbf{V}_{rel} \end{bmatrix} = \mathbf{H} \ \mathbf{J}_{ext} \begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} \tag{32}$$

The inverse kinematic control law with optimization function Z is defined as:

$$\begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} = (\mathbf{H} \ \mathbf{J}_{ext})^+ \begin{bmatrix} \dot{h} \\ V_{rel} \end{bmatrix} + \mathbf{P}Z \tag{33}$$

with $\mathbf{V_{rel}} = \mathbf{0}$ ensures the validation of the loop constraint. $\mathbf{P}$ is the projection into the null space of $\mathbf{HJ}_{ext}$ defined as:

$$\mathbf{P} = \left(\mathbf{I}_n - (\mathbf{H} \ \mathbf{J}_{ext})^+ (\mathbf{H} \ \mathbf{J}_{ext}) \right) \tag{34}$$

The minimum representation of task is implemented with different degrees of redundancy, by selecting an appropriate $\mathbf{H}$ matrix.

Point on plane A point on the object is moved to a plane by a series of elementary motions denoted as $\dot{h}$ along the normal to that plane. The task is defined by one direction whereas the *relative* variables contain six coordinates. Thus in total seven variables are controlled, maximizing the degrees of redundancy, while maintaining loop closure. The normal to the plane is used to project the velocity of the object to that directing towards the plane. The motion is uncontrolled in five directions but the normal projection guarantees that the point on the object will reach the plane. The orientation of the plane can be changed through the trajectory. The total kinematic equation is defined as:

$$\begin{bmatrix} \dot{h} \\ \mathbf{V}_{rel} \end{bmatrix} = \mathbf{H}_1 \mathbf{J}_{ext} \begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} \tag{35}$$

$$\mathbf{H}_1 = \begin{bmatrix} \mathbf{u}^T & \mathbf{0}_{1\times 9} \\ \mathbf{0}_6 & \mathbf{I}_6 \end{bmatrix} \tag{36}$$

$\mathbf{u}$ is the unit vector along the normal to the desired plane, for example to attain a horizontal plane:

$$\mathbf{u}^T = [0 \ 0 \ 1 \ ] \tag{37}$$

Point on Point A point on the object is moved to a point in space by a series of elementary motions with an arbitrary orientation. Three degrees of freedom of the object are controlled. Equation (32) is modified as follows:

$$\begin{bmatrix} \mathbf{v}_{tsk} \\ \mathbf{V}_{rel} \end{bmatrix} = \mathbf{H}_2 \mathbf{J}_{ext} \begin{bmatrix} \dot{\mathbf{q}}_r \\ \dot{\mathbf{q}}_l \end{bmatrix} \tag{38}$$

$$\mathbf{H}_2 = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times 9} \\ \mathbf{0}_6 & \mathbf{I}_6 \end{bmatrix} \tag{39}$$

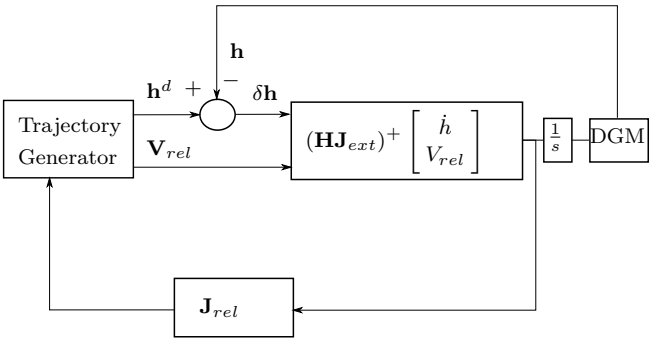

Fig. 3. Cooperative Control with Minimum Representation

The redundancy created by the minimum representation is employed to maximize the distance from the joint limits, utilizing the workspace as much as possible. The figures from Fig. 4 to 8 compare the joint values of the right arm, during a point to point trajectory. *Joint limit motion* is the solution when the joint limit optimization criterion is applied. *Pseudo* is the solution using the pseudoinverse solution.

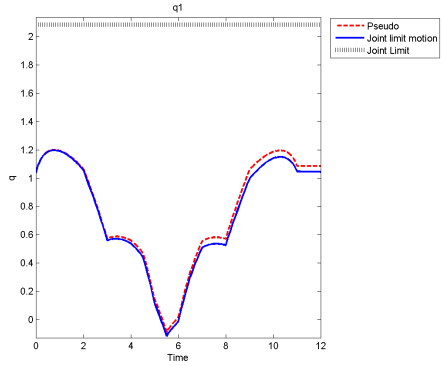

Fig. 4. Effect of joint limit optimization on joint q1

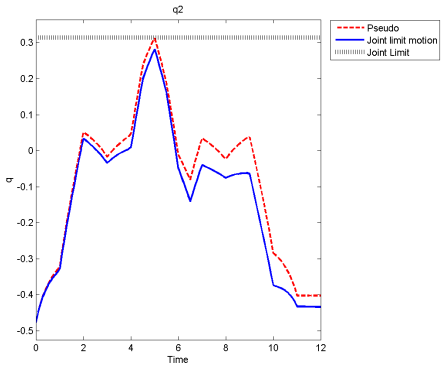

Fig. 5. Effect of joint limit optimization on joint q2

5.3 6 DOF Control of Object

The system described in table 1 is modified to include two additional passive revolute joints, the new M-DH parameters are given in table 2. These joints represent the object grasp conditions as shown in Fig.9 where the passive joint is denoted as q_{11} (q_{12} of the second chain). It

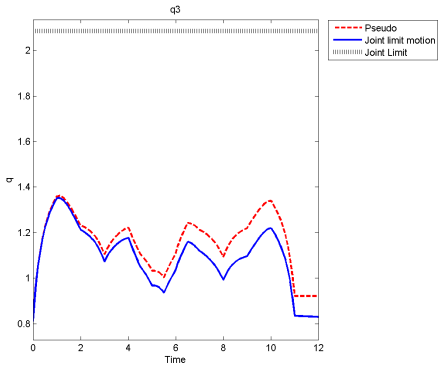

Fig. 6. Effect of joint limit optimization on joint q3

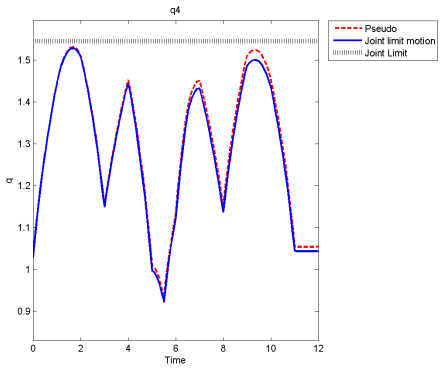

Fig. 7. Effect of joint limit optimization on joint q4

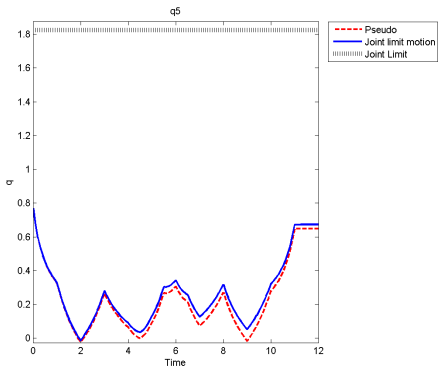

Fig. 8. Effect of joint limit optimization on joint q5

is assumed that the object grasp constrains five degrees of freedom but allows one degree of rotation about an axis that passes through the two contact points. By the addition of these passive joints the object gains 2-DOF. Thus $\mathbf{q}_a$ becomes a vector of six components. In contrast to Section 5.2, arbitrary forces can no longer be applied to the object since the chain contains passive joints. The parameters of frame 13 are defined by the grasp. The procedure defined in section 2 is carried out to find these parameters θ_{12} and θ_{11} represent the *passive* rotation of the aforementioned grasp condition.

Control By modeling the robot as a closed loop kinematic chain, the entire system can be controlled as one

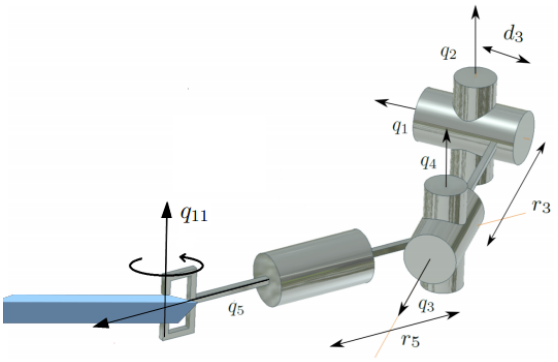

Fig. 9. Nao with passive joint

Table 2. MDH Parameters with passive grasp

j	$a(j)$	$\sigma(j)$	$\gamma(j)$	b	d	α	θ	r
1	0	0	0	b_1	0	$-\frac{\pi}{2}$	θ_1	$-r_1$
2	1	0	0	0	0	$\frac{\pi}{2}$	θ_2	0
3	2	0	0	0	$-d_3$	$\frac{\pi}{2}$	θ_3	r_3
4	3	0	0	0	0	$-\frac{\pi}{2}$	θ_4	0
5	4	0	0	0	0	$\frac{\pi}{2}$	θ_5	r_5
6	0	0	0	b_1	0	$-\frac{\pi}{2}$	θ_6	r_1
7	6	0	0	0	0	$\frac{\pi}{2}$	θ_7	0
8	7	0	0	0	d_3	$\frac{\pi}{2}$	θ_8	r_3
9	8	0	0	0	0	$-\frac{\pi}{2}$	θ_9	0
10	9	0	0	0	0	$\frac{\pi}{2}$	θ_{10}	r_5
11	5	0	0	0	d_{11}	0	θ_{11}	r_{11}
12	6	0	0	0	d_{12}	0	θ_{12}	r_{12}
13	11	2	γ_{13}	b_{13}	d_{13}	α_{13}	θ_{13}	r_{13}

dynamic entity. Fig. 10 illustrates a PID controller in task space, where K represents appropriate gain matrices.

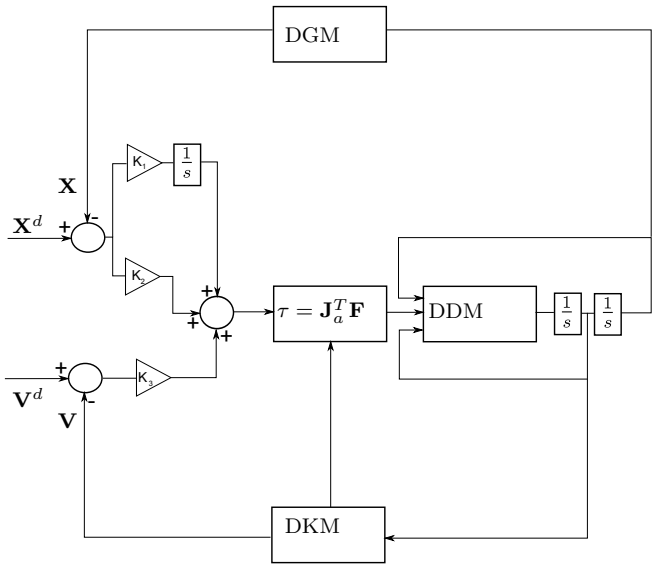

Fig. 10. PID Cooperative Control in Task Space

6. CONCLUSIONS

This work has presented a study of the two arms of the NAO robot when engaged in a cooperative task. By using cooperative task space formulation, an inverse kinematic controller with minimum representation of task is implemented. The minimum representation allows loop closure and also the creation a degree of redundancy. This redundancy is then employed to avoid the violation of joint limits. Secondly a task space PID is used to demonstrate the feasibility of the virtual passive joint. By including this passive joint two of the redundant actuators can be used to control the object directions in space.

REFERENCES

Bicchi, A., Melchiorri, C., and Balluchi, D. (1995). On the mobility and manipulability of general multiple limb robots. *IEEE Transactions on Robotics and Automation*, 11(2), 215–228.

Bonitz, R. and Hsia, T. (1996). Internal force-based impedance control for cooperating manipulators. *IEEE Transactions on Robotics and Automation*, 12(1), 78–89.

Caccavale, F., Chiacchio, P., and Chiaverini, S. (2000). Task-space regulation of cooperative manipulators. *Automatica*, 36(6), 879–887.

Caccavale, F., Chiacchio, P., Marino, A., and Villani, L. (2008). Six-DOF impedance control of dual-arm cooperative manipulators. *IEEE/ASME Transactions on Mechatronics,*, 13(5), 576–586.

Cheng, H., Yiu, Y.k., Member, S., and Li, Z. (2003). Dynamics and control of redundantly actuated parallel manipulators. *IEEE/ASME Transactions on Mechatronics*, 8(4), 483–491.

Chiacchio, P., Chiaverini, S., and Siciliano, B. (1996). Direct and inverse kinematics for coordinated motion tasks of a two-manipulator system. *Journal of Dynamic Systems, Measurement, and Control*, 118, 691.

Dombre, E. (1981). *Analyse des performances des robots-manipulateurs flexibles et redondants: contribution a leur modelisation et a leur commande*. Ph.D. thesis, USTL Montpellier, France.

Khalil, W. and Dombre, E. (2004). *Modeling, identification & control of robots*. Butterworth-Heinemann.

Khalil, W. and Kleinfinger, J. (1986). A new geometric notation for open and closed-loop robots. In *1986 IEEE International Conference on Robotics and Automation. Proceedings.*, volume 3, 1174–1179. IEEE.

Özkan, B. and Özgören, M. (2001). Invalid joint arrangements and actuator related singular configurations of a system of two cooperating scara manipulators. *Mechatronics*, 11(4), 491–507.

Sadati, N. and Ghaffarkhah, A. (2008). Decentralized impedance control of nonredundant multi-manipulator systems. In *IEEE International Conference on Networking, Sensing and Control, 2008. ICNSC 2008.*, 206–211.

Uchiyama, M. and Dauchez, P. (1988). A symmetric hybrid position/force control scheme for the coordination of two robots. In *1988 IEEE International Conference on Robotics and Automation, Philadelphia, PA*, 350–356.

Yeo, H.j., Suh, I.H., Yi, B.j., and Oh, S.r. (1999). A single closed-loop kinematic chain approach for a hybrid control of two cooperating arms with a passive joint: an application to sawing task. *IEEE Transactions on Robotics and Automation*, 15(1), 141–151.

Zielinski, C. and Szynkiewicz, W. (1996). Control of two 5 dof robots manipulating a rigid object. In *Proceedings of the IEEE International Symposium on Industrial Electronics, 1996. ISIE'96.,*, volume 2, 979–984. IEEE.