Task space control of grasped object in the case of multi-robots cooperating system

A. Khadraoui [a] **,** ***C. Mahfoudi*** [b] **,** ***A. Zaatri*** [c]**,** ***K. Djouani*** [d]

[a] *U. Larbi Ben M'Hidi , Institute of Mechanics, Ain Beida, Algeria*
Email: omar.khadraoui@gmail.com
[b] *U. Larbi Ben M'Hidi , Institute of Mechanics, Ain Beida, Algeria*
Email: c-mahfoudi- dz@yahoo.fr
[c] *U. Mentouri, Institute of Mechanics, Constantine,Algeria*
Email: azaatri@yahoo.com
[d] *LISSI Lab/University Paris 12, France*
Email: djouani@univ-paris12.fr

Abstract - This paper deals with multi-robots grasping problem. We present an optimal force distribution strategy for holding and manipulating objects by multiples manipulators. The force distribution issue is formulated in terms of a nonlinear programming problem under equality and inequality constraints. In particular, the friction constrains are transformed from non linear inequalities into a combination of linear equalities and linear inequalities. As a result, the original nonlinear constrained programming problem is then transformed into a quadratic optimization problem. The dynamical model of multi-robot cooperation has been used for determining force control distribution through proportional derivative (PD) controller. Simulation has been performed and some results are presented and discussed.

Keywords : dynamic modeling and control , friction constraints , grasping , Multi-robots cooperation , optimal force distribution , quadratic programming.

1 Introduction

Cooperative systems are generally understood as several coordinated robots simultaneously performing a common given task such as changing the space position of an object, grasping an object, gripping, lifting, lowering, releasing, withdrawing. The purpose of controlling a cooperative system consists of controlling contact forces between the environment and the object under consideration. Several approaches in the literature have been proposed to address the robot coordination problem. A robust control method is developed in [1] for a planar dual-arm manipulator system. In this approach, Contact and friction constraints for grasp conditions are considered and an optimization algorithm is developed which minimizes the total energy E consumed by the actuators. In [2], a robust force-motion control strategies are presented for mobile manipulators and force control constraints are developed using the passivity of hybrid joint rather than force feedback control. A new approach for computing force-closure grasps of two-dimensional and three-dimensional objects was proposed in [3],[4],[5], where a new necessary and sufficient conditions for n-finger grasps to achieve force-closure property are developed. In [6] a detailed analysis of grasping of deformable objects by a three finger hand was carried out, and it has been proved that the internal forces required for grasping deformable objects vary with size of object and finger contact angle. Xydas and al [7] have studied soft finger contact mechanics using finite element analysis and experiments. In [8], Hirai and al proposed a robust control law for grasping and manipulation of deformable objects. They developed a control law to grasp and manipulate a deformable object using a real time vision system. Cutkosky [9] showed that grasp stability is a function of the fingertip contact models and small changes in the grasp geometry. Mason and Salisbury [10] gave conditions for complete restraint of an object in terms of internal forces. Kerr and Roth [11] proposed a method to determine the optimal internal forces based on approximated frictional constraint at the object and fingertip. Meer and al [12],Patton and al [13] designed a system, which focuses on force control versus deformation control. In these systems the robot manipulator is designed to control the deformation of the object. In [14] Howard and Bekey developed a generalized learning algorithm for handling of 3-D deformable objects in which prior knowledge of object attributes is not required. They used neural network with mass, spring and damping constant as input and the force needed to grasp the object as output of the network.

The object force control needs of real-time calculation of force distribution on the robot's effectors. Due to the existence of more than three actuated joints in each robot, the manipulator system has redundant actuation leading to more active joints than the object degree-of-freedom (6 dof). Thus, when formulating the force distribution problem, we find fewer force moment balancing equations than unknown variables. So, the solution of these equations is not unique. Moreover, some physical constraints, that concern the contact nature, friction, ...etc , must be taken into account in the calculation of force distribution. In addition, joints torque saturation must also be considered. Thus the Force Distribution Problem (FDP) can be formulated as a nonlinear constrained programming problem under nonlinear equality and inequality constraints. In this context, several approaches have been proposed for solving such a problem,using mainly the following four principal methods :

1. Linear-Programming (LP) Method [15],[16]
2. Compact-Dual LP (CDLP) Method [17],[18]
3. Quadratic Programming (QP) Method [19], [20]
4. Analytical Method [21],[22],[23]

A comparative study for the four cited methods can be found in [24]. Some researchers proposed the optimal force distribution scheme of multiple cooperating robots by combining the Dual with the QP Method [25].

In this paper we propose an approach to solve the problem of real-time force distribution for multiple manipulators system grasping an object based on the approach proposed in [26],[27] for an hexapod robot. This approach consists of the combination of the QP method with the reduction technique of problem size. The main idea concerns the transformation of the original nonlinear constrained problem into a linear one, by reducing the problem size and transforming the nonlinear constraints into a linear ones, respecting some physical considerations. The rest of the paper is organized as follows: The direct and inverse geometrical models of the robot manipulator are presented in section 2. Section 3 concerns the force distribution problem. Problem reduction and optimal solution are presented in section 4. Before presenting some remarks and perspectives, a Matlab simulation results of two cooperating manipulators is presented in section 5 to show the efficiency of this approach.

2 Geometrical Modeling

Before presenting the direct and inverse geometrical model, let us consider the robot architecture. Since the robots are similar , only one robot modeling is considered. The dual-arm manipulator system holding a common object and its architecture is shown in figure(1) and figure(2). Every "j" robot j=1,...,n is grasping the object, which is located at a_j distance from the center of gravity of the object. The angle ϕ_j represents the orientation of the coordinate frame $(x_{1,j}, y_{1,j}, z_{1,j})$ fixed at the base of the robot and the world ground coordinate frame (X, Y, Z). Multiple manipulators system is considerated as an arborescent robot comporting some closed loops. So to study this kind of robots we use the method defined by Khalil and Kleinfinger [28].

The transformation matrix from ith joint's attached coordinate frame to the (i-1)th joint's attached coordinate frame is given by figure (3):

$$^{i-1}\boldsymbol{T}_i = R(Z,\gamma)T(Z,b)R(X,\alpha)T(X,d)R(Z,\theta)T(Z,r) \quad (1)$$

The table (1) describes the transformation from the world ground coordinate frame (X, Y, Z) to the coordinate frame at the contact point "6" of each robot"j", j=1,...,n

The transformation provides the exact position of the contact point"6" of each robot in the absolute

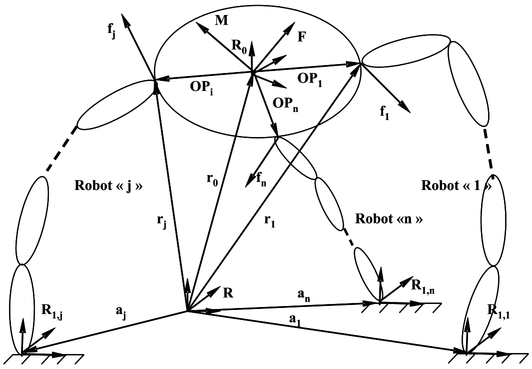

Figure 1: **Geometrical parameters of multiple manipulators system**

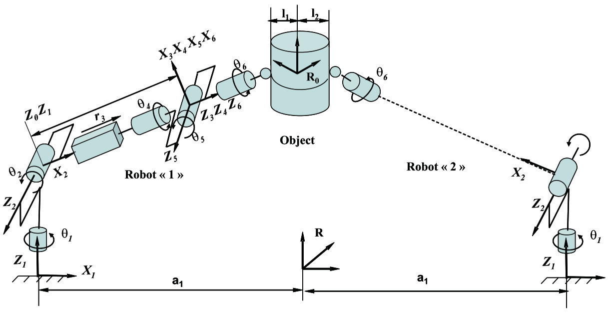

Figure 2: **Exemple of multiple manipulator system**

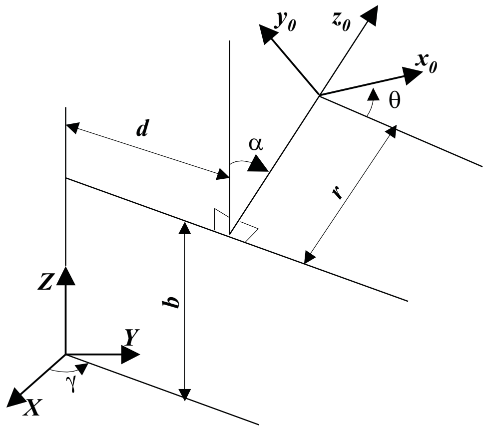

Figure 3: **Geometrical model**

coordinate frame fixed at the ground which is given by :

$$^{R}\boldsymbol{T}_6 = ^{R}\boldsymbol{T}_0^0 \boldsymbol{T}_1^1 \boldsymbol{T}_2^2 \boldsymbol{T}_3^3 \boldsymbol{T}_4^4 \boldsymbol{T}_5^5 \boldsymbol{T}_6 \quad (2)$$

When the position and the orientation of the last coordinate frame fixed to the end of each robot"j"

frame	α	d	θ	r	b	γ
object	α	d	θ	r	h	β
robot(1)	0	OP_1	$\theta_{1,1}$	0	0	0
robot"j"	0	OP_j	$\theta_{1,j}$	0	0	0
robot"n"	0	OP_n	$\theta_{1,n}$	0	0	0

Table 1: **Geometrical parameters of the system**

frame	α	d	θ	r	b	γ
joint"1"	0	a_j	θ_1	0	0	ϕ_j
joint"2"	0	0	$\theta_2 + \pi/2$	0	0	0
joint"3"	$\pi/2$	0	0	r_3	0	0
joint"4"	0	0	θ_4	r_4	0	0
joint"5"	$-\pi/2$	0	θ_5	0	0	0
joint"6"	$\pi/2$	0	θ_6	0	0	0

Table 2: **Geometrical parameters of the jth robot**

U_0 are known

$$U_0 = \begin{bmatrix} sx & nx & ax & Px \\ sy & ny & ay & Py \\ sz & nz & az & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

The values of the joints coordinates $\theta_{i,j}$ where : $\theta_{i,j}\ (i = 1, 2, 3, 4, 5, 6)\ (j = 1, .., n)$
are given as follow:

$$\begin{cases} \theta_{1,j} = \arctan(Py/(-Px + d_1)) \\ \theta_{2,j} = \arctan(Pz - r_1)/ \\ \quad (C1(Px - d_1) + S1Py)) \\ r_{3,j} = C2(C1(Px - d_1) + S1Py) + \\ \quad S2(Pz - r_1) - R4 - l_2 \\ \theta_{4,j} = \arctan((S1ax - C1ay)/ \\ \quad (S2(C1ax + S1ay) - C2az)) \\ \theta_{5,j} = \arctan((C4S2(C1ax - S1ay) - \\ \quad C4C2az + S4(C1ay - C1sy))/ \\ \quad (C2(C1ax + S1ay) + S2az)) \\ \theta_{6,j} = \arctan((-S4(-S2(C1ax - S1ay) + \\ \quad C2sz + C4S1sx - C1sy))/ \\ \quad (C5(-C4S2C1sx - S2S1sy + C2sz) \\ \quad +S4(S1sx - C1sy) - \\ \quad S5(C1C2sx + S1C2sy + S2sz)) \end{cases} \quad (4)$$

Remark : S*=sin(*); C*=cos(*); ($Px = P_{x,j}, Py = P_{y,j}, Pz = P_{z,j}$), are the coordinates of the point "6" of the jth robot expressed in (X, Y, Z). by using the Roulis, Tangage and Lacet (RTL) angles ,the desired position can be found as follow:

$$\begin{bmatrix} sx & nx & ax \\ sy & ny & ay \\ sz & nz & az \end{bmatrix} = \begin{bmatrix} C\phi C\theta & C\phi S\theta S\psi - S\phi C\psi \\ S\phi C\theta & S\phi S\theta S\psi + C\phi C\psi \\ -S\theta & C\theta S\psi \end{bmatrix}$$

$C\phi S\theta C\psi + S\phi C\psi$
$S\phi S\theta C\psi - C\phi S\psi$
$C\theta C\psi$.(5)
The Jacobian matrix developed above is sometimes called the geometric Jacobian to distinguish it from the analytic Jacobian, denoted by $Ja(q)$, which is based on a minimum representation for the orientation of the effector's coordinate frame $X = [x_p\, x_r]^T$. A representation of the situation of coordinate frame R_n in R_0, where x_p represents the three coordinates operational position and x_r is the coordinate operational orientation, the operational velocities are $\dot{X} = [\dot{x}_p\, \dot{x}_r]^T$ however we must find the relationship between these velocities and velocity vectors ${}^0\boldsymbol{V}_n\, {}^0\omega_n$ as follow :

$$\begin{bmatrix} \dot{x}_p \\ \dot{x}_r \end{bmatrix} = \begin{bmatrix} \Omega_p & 0_3 \\ 0_3 & \Omega_p \end{bmatrix} \begin{bmatrix} {}^0\boldsymbol{V}_n \\ {}^0\omega_n \end{bmatrix} = \Omega \begin{bmatrix} {}^0\boldsymbol{V}_n \\ {}^0\omega_n \end{bmatrix} \quad (6)$$

$$\begin{bmatrix} \dot{x}_p \\ \dot{x}_r \end{bmatrix} = \Omega^0 \boldsymbol{J}_n \dot{q} = \boldsymbol{J}_a \dot{q} \quad (7)$$

In general, the sub matrix Ω_p is equal to the unitary matrix I_3 because the operational coordinates of position are simply the Cartesian coordinates of the position of the tool. In contrast, the matrix Ω_r depends on the choice of the operational coordinates of rotation, as it is defined, the orientation vector we chose is $x_r = [\phi\, \theta\, \psi]^T$: representing the rotation called : roll, pith, and yaw (PRY). in this case [KHA 99]:

$$\Omega_r = \begin{bmatrix} C\phi tg\theta & S\phi tg\theta & 1 \\ -S\phi & C\phi & 0 \\ C\phi / C\theta & C\phi / C\theta & 0 \end{bmatrix}$$

we can show that,we have singularity when $\theta = +-\pi/2$.

3 Force Distribution Problem

3.1 Problem Formulation

The force system acting on the object is shown in figure (4). For simplicity, only the force components on the contact point are presented.

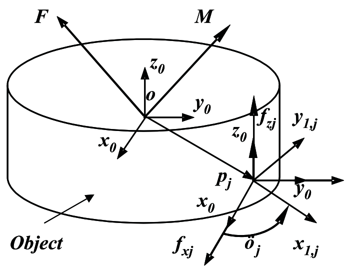

Figure 4: Orientation of coordinate frame

In the general case, rotational torques at the contact are neglected. Let (x_0, y_0, z_0) be the coordinate frame in which the object is located and $(x_{6,j}, y_{6,j}, z_{6,j})$ denote the coordinate frame fixed at the contact point of the jth robot. The $(x_{6,j}, y_{6,j})$ plane which is assumed to be parallel to the (x_0, y_0) plane and its z axis is normal to the surface of the object. $\boldsymbol{F} = [F_X F_Y F_Z]^T$ and $\boldsymbol{M} = [M_X M_Y M_Z]^T$ denote respectively the object force vector and moment vector, which results from the gravity and the external force acting on the object [29],[30],[31] and [32]. Define $f_{x,j}, f_{y,j}$, and $f_{z,j}$ as the components of the force acting on the supporting robot "j" in the directions of x_0, $y_0\, and\, z_0$, respectively. The number of cooperative robots, can vary between 2 and 3 for this studies. The object's quasi-static force/moment equation can be written as

$$\begin{cases} \sum_{j=1}^{n} \boldsymbol{f}_j = \boldsymbol{F} \\ \sum_{j=1}^{n} \boldsymbol{OP}_j \wedge \boldsymbol{f}_j = \boldsymbol{M} \end{cases} \quad (8)$$

where $\boldsymbol{OP}_j$ is the position vector joining contact point of the robot "j" and the gravity center of the object. The general matrix form of this equation can be written as:

$$\boldsymbol{A}\, \boldsymbol{G} = \boldsymbol{W} \quad (9)$$

with:

$$\begin{cases} \boldsymbol{G} = [\boldsymbol{f}_1^T\, \boldsymbol{f}_2^T\, \cdots\, \boldsymbol{f}_n^T]^T & \in \Re^{3n} \\ \boldsymbol{f}_j^T = [f_{x,j}\, f_{y,j}\, f_{z,j}]^T & \in \Re^3 \\ \boldsymbol{W} = [\boldsymbol{F}^T\, \boldsymbol{M}^T]^T & \in \Re^6 \end{cases}$$

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{I}_3 & \ldots & \ldots & \boldsymbol{I}_3 \\ \boldsymbol{B}_1 & \ldots & \ldots & \boldsymbol{B}_n \end{pmatrix} \quad \in \Re^{6 \times 3n}$$

$$\boldsymbol{B}_j \equiv \widehat{\boldsymbol{OP}}_j \equiv \begin{pmatrix} 0 & -P_{z,j} & P_{y,j} \\ P_{z,j} & 0 & -P_{x,j} \\ -P_{y,j} & P_{x,j} & 0 \end{pmatrix} \quad \in \Re^{3 \times 3}$$

where $\boldsymbol{I}_3$ is the identity matrix and $\mathbf{G}$ is the robots force vector, corresponding to three ($\boldsymbol{G} \in \Re^9$) .$\boldsymbol{A}$ is a coefficient matrix which is a function of the positions of the cooperative robots, and $\boldsymbol{B}_j$ is a skew symmetric matrix consisting of $(P_{x,j}, P_{y,j}, P_{z,j})$, which is the position coordinate of contact point of the multi-robot systems robot "j" in (x_0, y_0, z_0). $\boldsymbol{W}$ is a total body force/moment vector. It is clear that (9) is an undetermined system and its solution is not unique. In other words, the robots forces have many solutions according to the equilibrium equation. however, the robot forces must meet the needs for the following physical constraints, otherwise they become invalid :

1. Supported object should not slip when the robots move. It results in the following constraint:

$$\sqrt{f_{x,j}^2 + f_{y,j}^2} \leq \mu\, f_{z,j} \tag{10}$$

   where μ is the static coefficient of friction of the surface of the object

2. Since the robots forces are generated from the corresponding actuators of joints, the physical limits of the joint torques must be taken into account. It follows that :

$$-\tau_{jmax} \leq \boldsymbol{J}_j^{T\,j} \boldsymbol{A}_{0j} \begin{pmatrix} f_{x,j} \\ f_{y,j} \\ f_{z,j} \end{pmatrix} \leq \tau_{jmax} \tag{11}$$

   for $(j = 1, ..., n)$, where $\boldsymbol{J}_j \in \Re^{3\times6}$ is the Jacobian of the robot "j", $\tau_{jmax} \in \Re^{6\times1}$ is the maximum joint torque vector of the robot "j", and $\boldsymbol{A}_{0j} \in \Re^{3\times3}$ is the orientation matrix of $(x_{6,j}, y_{6,j}, z_{6,j})$ with respect to (x_0, y_0, z_0).

3. In order to have definite contact with the object, there must exist a $f_{z,j}$ such that :

$$f_{z,j} \geq 0 \tag{12}$$

   In the following, we propose an approach for problem size reduction, linearization and solving for the three manipulators case. Clearly, it is difficult to solve such a nonlinear programming problem for real-time multi-robots force distribution with complex constraints.

3.2 Problem Size Reduction

The equation (10) is a formulation of the friction cone figure(5). In order to overcome the non linearities induced by the following equations, most researches substitute this friction cone by the inscribed pyramid [16],[33],[19],[34]. Thus, the nonlinear friction constraints are approximately expressed by the following linear inequalities :

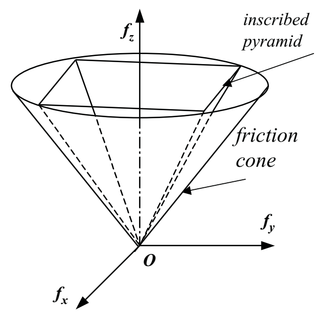

Figure 5: conservative inscribed pyramid

$$f_{x,j} \geq \acute{\mu}\, f_{z,j}, \quad f_{y,j} \geq \acute{\mu}\, f_{z,j}, \quad j = 1, ..., n \tag{13}$$

where $\acute{\mu} = \frac{\sqrt{2}\mu}{2}$ is for the inscribed pyramid.

Thus, the initial non linear constrained programming problem, substituting the non linear constraint Eq(10) by the linear one of Eq(13), becomes a linear programming problem [16],[17], and [19]. The possibility of slipping can be minimized, by optimizing the ratio of tangential to normal forces at the robot. This leads Liu and Wen [22] to find the relationship between the robot forces and transform the initial friction constraints from the nonlinear inequalities into a set of linear equalities. Let us define the global ratio by the ratio of the tangential to normal forces at the object. The advantage of the existing methods lies in the fact that part of component of the robots forces satisfy the global ratio relation ship and lets the other components satisfy the linear inequality constraints as Eq (13). For example, defining $f_{x,j} \quad (j = 1, ..., n)$ and $f_{y,j} \quad (j = 1, ..., n)$, for a robot j. Chen et al [1], show that :

$$f_{x,j} = k_{xz} f_{z,j} \quad , (i = 1, ..., n) \tag{14}$$

$$f_{y,j} \leq \mu^{\star} f_{z,j} \quad , (i = 1, ..., n) \tag{15}$$

where $k_{xz} = \frac{F_X}{F_Z}$ is the global ratio of forces at the object in direction of x_0 and z_0. $\mu^{\star}$ is the given coefficient for friction constraints. According to Eq(10), we have $\mu^{\star} = \sqrt{\mu^2 - k_{xz}^2}$. Finally, the force distribution problem is transformed into a linear one by replacing Eq (6) with Eqs (10) and (11).

3.3 Problem transformation and Continuous solution

In modelling these systems, we consider that three robots grasp the object at a time. so $\boldsymbol{G}$ and $\boldsymbol{A}$ become a vector of 9×1 and a matrix of 6×9, respectively. Expression (9) contains nine unknown variables with six equations. By adding the Eq (14) to the Eq (9) we obtain nine equations.

$$\overline{\boldsymbol{A}}\,\boldsymbol{G} = \overline{\boldsymbol{W}} \tag{16}$$

with :

$$\overline{\boldsymbol{A}} = \begin{pmatrix} & \boldsymbol{I}_3 & & & \boldsymbol{I}_3 & & & \boldsymbol{I}_3 & \\ & \boldsymbol{B}_1 & & & \boldsymbol{B}_2 & & & \boldsymbol{B}_3 & \\ 1 & 0 & -k_{xz} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -k_{xz} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -k_{xz} \end{pmatrix}$$

$$\boldsymbol{G} = \begin{pmatrix} \boldsymbol{f}_1 \\ \boldsymbol{f}_2 \\ \boldsymbol{f}_3 \end{pmatrix} \qquad \overline{\boldsymbol{W}} = \begin{pmatrix} \boldsymbol{F} \\ \boldsymbol{M} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Using some rows combination of the matrix $\overline{\boldsymbol{A}}$, Eq (16) can be written as :

$$\widehat{\boldsymbol{A}}\,\boldsymbol{G} = \widehat{\boldsymbol{W}} \tag{17}$$

With: $\widehat{\boldsymbol{A}} =$

$$\left(\begin{array}{ccccccccc} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & -k_{xz} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -k_{xz} & 0 & 0 & 0 \\ -P_{y,1} & P_{x,1} & 0 & -P_{y,2} & P_{x,2} & 0 & 1 & 0 & -k_{xz} \\ P_{z,1} & 0 & -P_{x,1} & P_{z,2} & 0 & -P_{x,2} & -P_{y,3} & P_{x,3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & P_{z,3} & 0 & -P_{x,3} \\ 0 & -P_{z,1} & P_{y,1} & 0 & -P_{z,2} & P_{y,2} & 0 & -P_{z,3} & P_{y,3} \end{array}\right) \qquad \widehat{\boldsymbol{W}} = \begin{pmatrix} F_X \\ F_Y \\ 0 \\ 0 \\ M_Z \\ M_Y \\ 0 \\ M_X \end{pmatrix}$$

where $\widehat{\boldsymbol{A}} \in \Re^{8\times 9}$ is the resulting matrix of $\overline{\boldsymbol{A}}$ after combination. $\boldsymbol{G} \in \Re^{8}$ is the robots force vector. $\widehat{\boldsymbol{W}} \in \Re^{8}$ is the resulting vector of $\boldsymbol{W}$ after combination. Thus, the force distribution problem is subjected to the inequality constraints expressed by (11), (12) (14).

4 Quadratic Problem Formulation and Solution

The solution to the inverse dynamic equations of this system is not unique, but it can be chosen in an optimal manner by minimizing some objective functions. The approach taken here consists of minimizing the sum of the weighted torque of the robot, which results in the following objective function [20],[25],[35] :

$$f_{\boldsymbol{G}} = \boldsymbol{p}^T \boldsymbol{G} + \frac{\boldsymbol{G}^T \boldsymbol{Q} \boldsymbol{G}}{2} \tag{18}$$

with:

$$\boldsymbol{p}^T = [\widehat{\tau}_1^T \boldsymbol{J}_1^T \,\ldots\ldots, \widehat{\tau}_n^T \boldsymbol{J}_n^T] \quad \in \Re^{3n}$$

$$\boldsymbol{Q} = \begin{pmatrix} \boldsymbol{J}_1 \boldsymbol{q}_1 \boldsymbol{J}_1^T & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \boldsymbol{J}_n \boldsymbol{q}_n \boldsymbol{J}_n^T \end{pmatrix} \quad \in \Re^{3n\times 3n}$$

where $\widehat{\tau_j}$ is the joint torque vector due to the weight and inertia of the robot "j", $\mathbf{J}_j$ is the Jacobian of the robot "j", and $\mathbf{q}_j$ is a positive definite diagonal weighting matrix of the robot"j". This objective function is strictly convex. Because the time for obtaining a solution does not depend on an initial guess, a quadratic programming is superior to linear programming in both speed and quality of the obtained solution [20]. The general linear-quadratic programming problem of the force distribution on robot is stated by :

$$\boldsymbol{p}^T \boldsymbol{G} + \frac{\boldsymbol{G}^T \boldsymbol{Q} \boldsymbol{G}}{2} \tag{19}$$

$$\widehat{\boldsymbol{A}}\,\widehat{\boldsymbol{G}} = \widehat{\boldsymbol{W}} \tag{20}$$

$$\boldsymbol{B}\widehat{\boldsymbol{G}} \leq \boldsymbol{C} \tag{21}$$

where $\mathtt{G} \in \Re^9$ is a vector of the design variables. It should be pointed out that the expression (21) is the resulting inequality constraints for the combination of equations (11),(12) and (15) where :

$$\boldsymbol{B} = [\boldsymbol{B}_1^T\, \boldsymbol{B}_2^T\, \boldsymbol{B}_3^T\, \boldsymbol{B}_4^T]^T \quad \in \Re^{9\times 24}$$
$$\boldsymbol{C} = [\tau_{1max} \,..\, \tau_{6max} \;\; -\tau_{1max} \,..\, - \tau_{6max}\; \boldsymbol{0}\,\boldsymbol{0}\,\boldsymbol{0}\,\boldsymbol{0}\,\boldsymbol{0}\,\boldsymbol{0}]^T \quad \in \Re^{24}$$

with

$$B_1 = \begin{bmatrix} \boldsymbol{J}_1^T\boldsymbol{R}_1 & 0 & 0 \\ 0 & \boldsymbol{J}_2^T\boldsymbol{R}_2 & 0 \\ 0 & 0 & \boldsymbol{J}_3^T\boldsymbol{R}_3 \end{bmatrix} \quad \in \Re^{9\times 9}$$

$$\boldsymbol{B}_2 = \begin{bmatrix} -\boldsymbol{J}_1^T\boldsymbol{R}_1 & 0 & 0 \\ 0 & -\boldsymbol{J}_2^T\boldsymbol{R}_2 & 0 \\ 0 & 0 & -\boldsymbol{J}_3^T\boldsymbol{R}_3 \end{bmatrix}$$

$$B_3 = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$B_4 = \begin{bmatrix} 0 & 1 & -\mu^\star & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -\mu^\star & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\mu^\star \end{bmatrix}$$

In Eq (20), we have eight linear independent equations with nine unknown variables. By using Gauss algorithm, this equation is transformed as follow :

$$[\boldsymbol{I}_8 \quad \widehat{\boldsymbol{A}}_r] \begin{bmatrix} \widehat{\boldsymbol{G}}_b \\ \widehat{\boldsymbol{G}}_r \end{bmatrix} = \widehat{\boldsymbol{W}}_r \tag{22}$$

where $\boldsymbol{I}_8 \in \Re^{8\times 8}$ identity matrix, $\widehat{\boldsymbol{A}}_r \in \Re^8$ is the remaining column of the matrix $\widehat{\boldsymbol{A}}$ after transformation. $\widehat{\mathbf{G}}_b \in \Re^8$ is the partial vector of $\boldsymbol{G}$. $\widehat{\boldsymbol{G}}_r \in \Re$ is the unknown element of $\widehat{\boldsymbol{G}}$ which denotes the design variable. $\widehat{\boldsymbol{W}}_r \in \Re^8$ is the resulting vector of $\widehat{\boldsymbol{W}}$ after transformation. Equation (22) may be rewritten by the following form

$$\boldsymbol{I}_8\widehat{\boldsymbol{G}}_b + \widehat{\boldsymbol{A}}_r\widehat{\boldsymbol{G}}_r - \widehat{\boldsymbol{W}}_r = 0, \tag{23}$$

Which yields to

$$\widehat{\boldsymbol{G}}_b = \widehat{\boldsymbol{W}}_r - \widehat{\boldsymbol{A}}_r\widehat{\boldsymbol{G}}_r. \tag{24}$$

Finally, it results in

$$\boldsymbol{G} = \begin{bmatrix} \widehat{\boldsymbol{G}}_b \\ \widehat{\boldsymbol{G}}_r \end{bmatrix} = \begin{bmatrix} \widehat{\boldsymbol{W}}_r \\ 0 \end{bmatrix} + \begin{bmatrix} -\widehat{\boldsymbol{A}}_r \\ 1 \end{bmatrix} \widehat{\boldsymbol{G}}_r. \tag{25}$$

Now let $\widehat{\boldsymbol{G}}_0 = [\widehat{\boldsymbol{W}}_r^T \quad 0]^T \in \Re^8$ *and* $N = [-\widehat{\boldsymbol{A}}_r^T \quad 1]^T \in \Re^9$, then Eq (25) becomes

$$\boldsymbol{G} = \widehat{\boldsymbol{G}}_0 + \boldsymbol{N}\widehat{\boldsymbol{G}}_r. \tag{26}$$

Substituting Eq (26) into Eqs (19) and (21), the linear quadratic programming problem can be expressed by :

$$minimize \quad f(\widehat{\boldsymbol{G}}_r), \tag{27}$$

$$subject\,to \quad \boldsymbol{B}\boldsymbol{N}\widehat{\boldsymbol{G}}_r \le \boldsymbol{C} - \boldsymbol{B}\widehat{\boldsymbol{G}}_0. \tag{28}$$

where

$$f(\widehat{\boldsymbol{G}}_r) = \boldsymbol{p}^T\widehat{\boldsymbol{G}}_0 + \tfrac{1}{2}\widehat{\boldsymbol{G}}_0^T Q\widehat{\boldsymbol{G}}_0 + \boldsymbol{p}^T\boldsymbol{N}\widehat{\boldsymbol{G}}_r + \tfrac{1}{2}\widehat{\boldsymbol{G}}_0^T\boldsymbol{Q}\boldsymbol{N}\widehat{\boldsymbol{G}}_r + \tfrac{1}{2}\widehat{\boldsymbol{G}}_r^T\boldsymbol{N}^T\boldsymbol{Q}\widehat{\boldsymbol{G}}_0 + \tfrac{1}{2}\widehat{\boldsymbol{G}}_r^T\boldsymbol{N}^T\boldsymbol{Q}\boldsymbol{N}\widehat{\boldsymbol{G}}_r$$

Since $\widehat{\boldsymbol{G}}_r$ is a single variable denoted by x, the optimal force distribution can be written as :

$$minimize \quad a_0x^2 + a_1x + a_2 \qquad subject\,to \quad x \in [b_1\, b_2] \tag{29}$$

With

$$a_0 = \tfrac{1}{2}\boldsymbol{N}^T\boldsymbol{Q}\boldsymbol{N}$$
$$a_1 = \boldsymbol{p}^T\boldsymbol{N} + \tfrac{1}{2}\widehat{\boldsymbol{G}}_0^T\boldsymbol{Q}\boldsymbol{N} + \tfrac{1}{2}\boldsymbol{N}^T\boldsymbol{Q}\widehat{\boldsymbol{G}}_0$$
$$a_2 = \boldsymbol{p}^T\widehat{\boldsymbol{G}}_0 + \tfrac{1}{2}\widehat{\boldsymbol{G}}_0^T\boldsymbol{Q}\widehat{\boldsymbol{G}}_0$$

Where $[b_1\, b_2]$ denotes the bound resulted from Eq(28). Since it is clear that $a_0 \ge 0$ because of the positive-definite matrix Q, There must be an optimal solution for the force distribution problem.

5 Robot dynamic modeling

For each robot, the serial robot dynamic model [28], [36], is given by the following equations :

$$\Gamma = f(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) = \boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{J}^T\boldsymbol{f} \tag{30}$$

$\boldsymbol{C} \in \Re^{6\times 1}$, $\boldsymbol{Q} \in \Re^{6\times 1}$ *and* $\boldsymbol{J}^T\boldsymbol{f} \in \Re^{6\times 1}$.Where,

- $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$ and $\ddot{\boldsymbol{\theta}}$ are, respectively, the generalized coordinate, speed and acceleration vectors,
- $\boldsymbol{M}(\boldsymbol{\theta})$, $n \times n$ matrix representing the inertia of the robot,
- $\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$, $n \times 1$ vector representing coriolis and centrifuge terms,
- $\boldsymbol{Q}(\boldsymbol{\theta})$, $n \times 1$ vector of gravity terms
- $\boldsymbol{J}^T$, jacobian matrix of the robot
- $\boldsymbol{f}$, force acting on the supporting robot "j"

Equation 30 can be written as follows :

$$\Gamma = \boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{H}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{J}^T \boldsymbol{f} \tag{30}$$

with :

$$\boldsymbol{H}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \boldsymbol{Q}(\boldsymbol{\theta})$$

By setting $\ddot{\boldsymbol{\theta}}$ and $\boldsymbol{f}$ to $\boldsymbol{0}$, we can compute $\boldsymbol{H}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ using Newton-Euler algorithm.

5.1 Object Dynamic

Consider an object of mass m_0 being carried by the two manipulators, so when the load is grasped by the end-effectors of robots , the force exerted from the tips of the manipulators act to translate or rotate the object in any direction, then the dynamic model of the object is given by :

$$\begin{pmatrix} m_0\boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \underline{\boldsymbol{\varphi}}^0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\gamma}_{0,0} \\ \dot{\boldsymbol{\omega}}_{0,0} \end{pmatrix} + \begin{pmatrix} -m_0\mathbf{g}_0 \\ \boldsymbol{\omega}_{0,0} \wedge (\underline{\boldsymbol{\varphi}}^0 \boldsymbol{\omega}_{0,0}) \end{pmatrix} = \begin{pmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{pmatrix} \tag{31}$$

$\boldsymbol{F}$ and $\boldsymbol{M}$ are given by the following equations.

$$\begin{cases} \boldsymbol{F} = \sum_{j=1}^{6} \boldsymbol{F}_{1,j}^0 \\ \boldsymbol{M} = \sum_{j=1}^{6} (\boldsymbol{C}_{1,j}^0 + \boldsymbol{P}_{01,j}^0 \wedge \boldsymbol{F}_{1,j}^0) \end{cases} \tag{32}$$

where :

- $\boldsymbol{\gamma}_{0,0}$ and $\boldsymbol{\omega}_{0,0}$ be, respectively, the object linear and angular accelerations in the coordinate frame (x_0, y_0, z_0).
- $\boldsymbol{F}_{1,j}^0$, the force applied by the robot "j" at the articulation "6" on the object "0".
- $\boldsymbol{C}_{1,j}^0$, the moment applied by the j^{th} robot in the articulation "6" on the object.
- $\boldsymbol{P}_{01,j}^0$, the distance between the articulation "6" of the j^{th} robot and the origin of the coordinate frame (x_0, y_0, z_0) expressed in the same coordinate frame.
- m_0 and $\underline{\boldsymbol{\varphi}}^0$ are, respectively, the mass and the inertia matrix of the grasped object.
- $\boldsymbol{I}_3$, the identity matrix (3×3).
- $\mathbf{g}_0$: gravity vector

6 Systems Dynamic Joint Control

Let us define $X_d(t)$, the desired trajectory for the arm motion. To ensure trajectory tracking by the joint variable errors and the optimal force distribution algorithm, computed torque control is used for robots joint level control (figures 6 and 7).

with;

$$X_0^d = [P_{x0}, P_{y0}, P_{z0}, \theta_0, \phi_0, \psi_0]$$
$$X_1^d = [P_{x1}, P_{y1}, P_{z1}, \theta_1, \phi_1, \psi_1]$$
$$X_2^d = [P_{x2}, P_{y2}, P_{z2}, \theta_2, \phi_2, \psi_2]$$

where:

X_0^d, X_1^d and X_2^d denote respectively the desired p osition and orientation of object, robot1 and robot2

$$\boldsymbol{e}(t) = X_d - X(t) \tag{33}$$

Then, the overall robot arm input becomes:

$$\Gamma = \boldsymbol{A}\boldsymbol{J}^{-1}(\ddot{\boldsymbol{X}} - \dot{\boldsymbol{J}}\dot{\theta}) + \boldsymbol{H} \tag{34}$$

$$\boldsymbol{X}(t) = \ddot{\boldsymbol{X}} + \boldsymbol{k}_v(\dot{\boldsymbol{X}}_d - \dot{\boldsymbol{X}}) + \boldsymbol{k}_p(\boldsymbol{X}_d - \boldsymbol{X}) \tag{35}$$

This controller is shown in figures (6) and (7)

6.1 Choice of PD Gains

It is usual to take the $n \times n$ matrices diagonal so that:

$$\boldsymbol{k}_v = diag[k_{vi}], \quad \boldsymbol{k}_p = diag[k_{pi}]$$

and $k_{pi} = \omega_n^2$, $k_{pi} = 2\xi\omega_n$ with ξ the damping ratio and ω_n the naturel frequency.
The PD gains are usually selected for critical damping ξ=1. Then, to avoid exciting the resonant mode, we should select natural frequency to half the resonant frequency $\omega_n < \omega_r/2$.

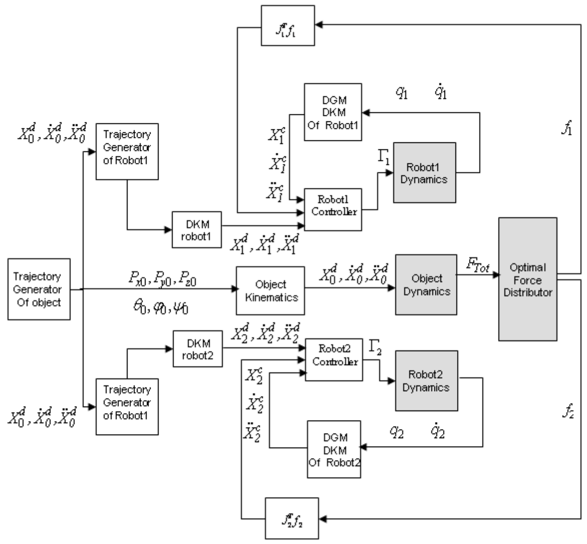

Figure 6: **System overall control**

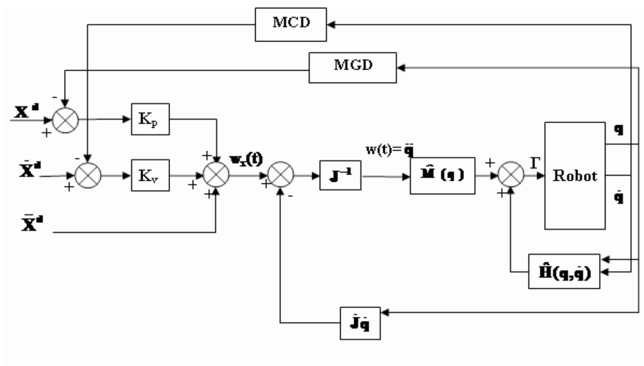

Figure 7: **Torque control of the robot** $"j"$

7 Simulations

In order to show the effectiveness of proposed approach, we consider two identical 6-DOF manipulators handling a rigid object in a task space. The base of each robot is located at a distance a_1 see figure(2), We denote $X_i \in \Re^6$ where: $X_i=[P_{xi}, P_{yi}, P_{zi}, \theta_i, \phi_i, \psi_i]^T$ The end-effector position of robot i is designed by: $[P_{xi}, P_{yi}, P_{zi}]^T$, while the orientation is represented by $[\theta_i, \phi_i, \psi_i]^T$. Furthermore the load vector coordinates (generalized forces) at the object is known. The values of masse $m(i)$ in Kg, the moment of inertia $Ix(i)$, $Iy(i)$,$Iz(i)$ in $Kg.m^2$ and the lenghts in meter of different links of each robot used in simulation , are given in table (3) and $\mu = 0.05$ is the static coefficient of friction.

Link"i"	$Ix(i)$	Iy(i)	$Iz(i)$	m(i) Kg	l(i)
Link"1"	0.01	0.05	0.06	25	1
Link"2"	0.1	0.2	0.3	30	0.75
Link"3"	0.4	0.4	0.2	15	0.50
Link"4"	0.3	0.5	0.01	4	0.25
Link"5"	0.01	0.6	0.03	4	0.10
Link"6"	0.2	0.2	0.6	2	0.1

Table 3: **Physical parameters of the robot**

Some simulations were conducted with Matlab. We consider that the desired trajectory of the object in the operational space is given as follows : $x = 1 + 2.5.sin(0.3)t$, $y{=}{-}3 + 2.t$, $z{=}r_{obj}{=}r_1 + sin(2.t)$, $\theta(t)_z{=}\pi/2 + 1.5.sin(t)$ is the rotation about the ox axis and $\alpha_x(t){=}\ \pi/2{+}0.5.sin(0.3t)$ the rotation about the oy axis. $\tau_j = [10\,10\,10\,10\,10\,10]^T$is the maximum joint torque vector and For the objective function Eq (18), the weighting matrix are choosen as follow : $\boldsymbol{p} = \boldsymbol{0}$and $\boldsymbol{Q} = \boldsymbol{I}$ (the identity matrix).
The figure 8 shows the trajectory of the object in the X-Y-Z supported by the two robots.

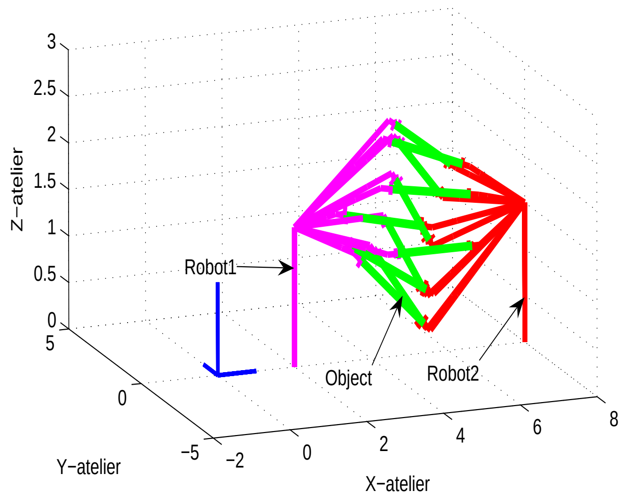

Figure 8: View of the object suported by tow robots

The associated joints coordinates are obtained by using the direct and inverse geometric model (Eq (2)and Eq (3)). The Figure (9.a) and (9.b) show respectively the profiles of the position P_x,

P_y, P_z and orientation θ, ϕ, ψ of the grasped object in the operational space.

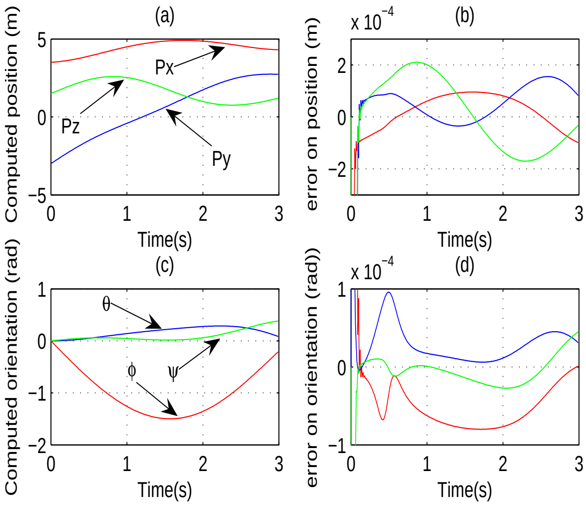

Figure 9: Computed position and orientation of robot1

However the position P_x, P_y, P_z and orientation θ, ϕ, ψ respectively of robot1 and robot2 are presented in Figures (10.a),(10.c) and (11.a),(11.c). The figures (10.b),(10.d), (11.b),(11.d),(12.c),(12.d) and (13.c),(13.d) demonstrates convergence of corresponding signals illustrated in figures cited above. In the figures (14.a) and (14.b) the force distribution acting respectively by robot(1) and robot(2) on the grasped object are given. We can show that, this distribution validate the following force equilibrium equation :

$$\Sigma f x_j = Fx, \quad ,\Sigma f y_j = Fy, \quad \Sigma f z_j = Fz$$

Elsewhere, the z-force components fz_j are never negative, respecting the contact constraint . We can also show that the constraint Eq (10) are always satisfied as shown in figure (14c). and (14d) for the robot(j) ,j=1,2 where the curve $fx_j^2 + fy_j^2$ is always under the curve $\mu^2 fz_j^2$

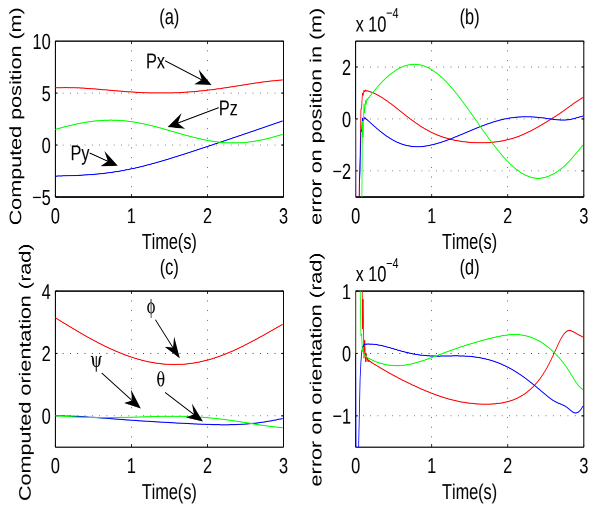

Figure 10: Computed position and orientation of robot2

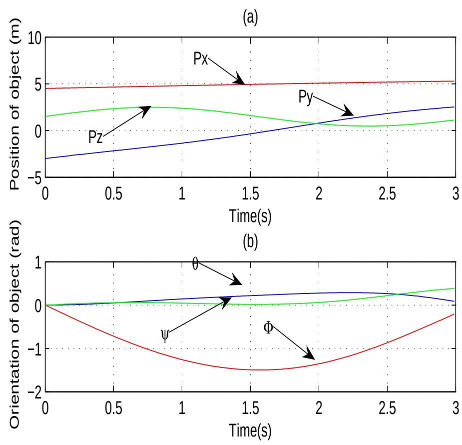

Figure 11: Position and orientation of the grasped object

8 Conclusion

In this paper, the force distribution problem has been presented in the case of multiple manipulators system grasping a same object. First, the robot inverse and direct geometric models are presented. Then, the real time force distribution problem are formulated in terms of non linear programming problem. After problem size reduction and transformation, the initial problem is solved in terms of quadratic programming problem. After then, we present the development for real-time operational space control of the system.Finally, Simulations results are presented in order to show the effectiveness of the

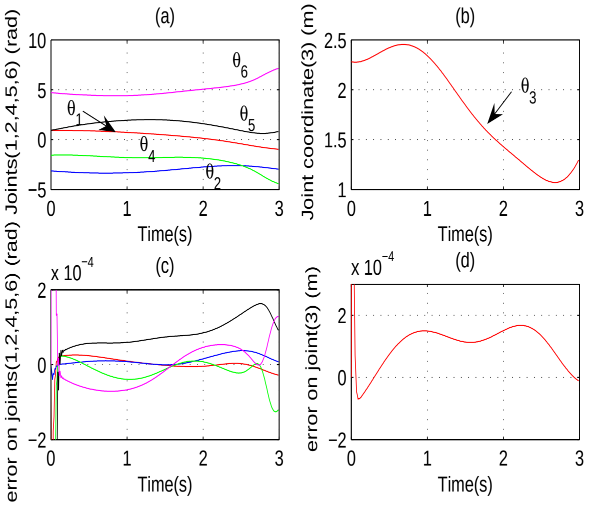

Figure 12: Joints coordinates of the robot2

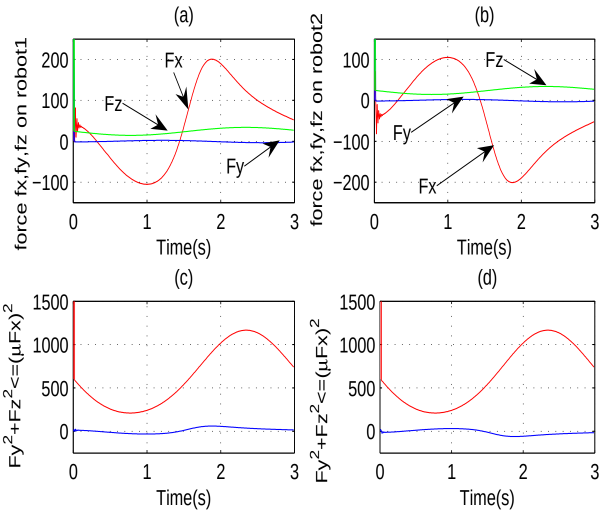

Figure 14: Forces on the two robots

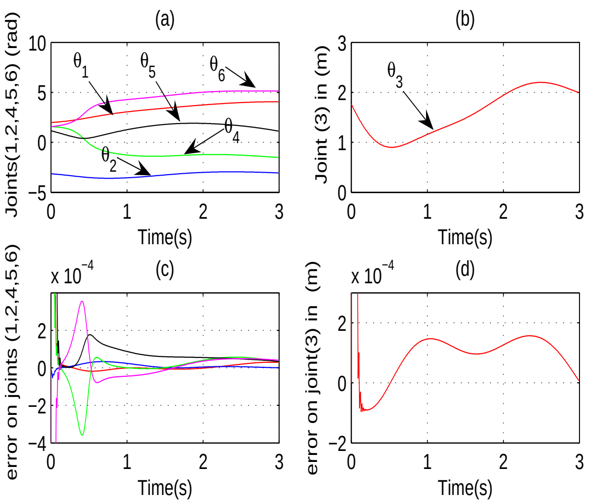

Figure 13: Joints coordinates of the robot1

proposed approach. Actually, we are working on the generalization of this approach for $n > 3$ robots case.

References

[1] Jin-Fan Liu Karim Abdel-Malek1. Robust control of planar dual-arm cooperative manipulators. *Robotics and Computer-Integrated Manufacturing.*, pages 109–120, (2000).

[2] Zhijun Li1*; Weidong Chen1 and Hong Liu2. Robust control of wheeled mobile manipulators using hybird joints. *International Journal of Advanced Robotic Systems*, 5:83–90, (2008).

[3] B.Bounab D.Sidobre and A.Zaatri. On computing multi-finger force closure grasps of 2d objects. *ICINCO Int. Conf. Informatics in Control, Automat. Robot.*, (2007).

[4] B.Bounab D.Sidobre and A.Zaatri. Central axis approach for computing n-finger force-closure grasp. *in Proc. IEEE Int. Conf. Robot. Automat.*, pages 1169–1174, (2008).

[5] N.Saito T.Satoh and H.Okano. Grasping force control in consideration of translational and rotational slippage of an object by a flexible contact sensor. *Electrical Engineering in Japan*, 165:51–57, 2008.

[6] Sohil Garg and Ashish Dutta. Grasping and manipulation of deformable objects based on internal force requirements. *International Journal of Advanced Robotic Systems*, 3:107–114, (2006).

[7] Xydas. Study of soft finger contact mechanics using finite element analysis and experiments. *Proc. of the IEEE Intl. cong on robotics and Automation*, 3:2179 –2184, 2000.

[8] Hirai.S Tsuboi.T and Wada .T. Robust grasping manipulation of deformable objects. *IEEE Symposium on Assembly and*

Task Planning.Japan, pages 2179 –2184, 2001.

[9] M.R. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *Trans Robot. Automat*, 5:269–279, 1989.

[10] Mason and J.K. Salisbury. Robot hands and the mechanics of manipulation. *Cambridge, MA: MIT Press*, 5, 1985.

[11] J. Kerr and B. Roth. Analysis of multifingered hands. *International Journal of Robotics Research*, 4:3–17.

[12] D.W. Meer and S.M. Rock. Experiments in object impedance control for flexible objects. *Proceedings International Conference on Robotics and Automation*, 4:1222–1227.

[13] S. Tricamo R. Patton, F. Swern and A. van der Veen. Automated cloth handling using adaptive force feedback. *Journal of Dynamic Systems and Measurements and Control*, 114:731–733.

[14] A. M. Howard and G.A. Bekey. Intelligent learning for deformable object manipulation.

[15] D. E. Orin and S. Y. Oh. Control of force distribution in robotic mechanisms containing closed kinematic chains. *" Trans. of the ASME, J. of Dynamic Systems, Measurement, and Control*, 102:134–141, 1981.

[16] C. A. Klein and S. Kittivatcharapong. Optimal force distribution for the legs of a walking machine with friction cone constraints. *IEEE Trans. on Robotics and Automation*, 6:73–85, 1990.

[17] F. T.Cheng and D. E. Orin. Efficient formulation of the force distribution equations for simple closed-chain. *IEEE Trans. on Systems, Man, and Cybernetics*, 21:25–32, 1991.

[18] J.Cornella R.Suarez and R.Carloni. Dual programming based approach for optimal grasping force distribution. *" Mechatronics (Oxford)*, 165:348–356, 2008.

[19] Nahon.MA.and Angeles.J. Optimization of dynamic forces in mechanical hands. *J. Mech. Trans. Autom. Design*, pages 167–173, 1991.

[20] M. A. Nahon and J. Angeles. Real-time force optimization in parallel kinematic chains under inequality constraints. *IEEE Trans. on Robotics and Automa.*, 12:87–94, 1999.

[21] KOO Patrick. An analytical expression for the generalized forces in multibody lagrange equations. *IEEE transactions on robotics and automation*, 20:340–343, 2004.

[22] H. Liu and B. Wen. Force distribution for the legs of quadruped robot. *Journal of Robotique Systemes*, 14:1–8, 1997.

[23] G.Lin D.Wang and LI. The analytical dynamic model of six-dof industrial robotic manipulators of containing closed chain. *Mechanism and machine theory*, 40:385–393, 2005.

[24] X. Chen and K.Watanabe. Optimal force distribution for the legs of quadruped robot. *Machine inteligence and robotique control*, 1:87–94, 1999.

[25] W. Kwon and B.C. Wen. A new optimal force distribution scheme of multiple cooperating robots using dual method. *J. of Intelligent and Robotic Systems*, 21:301–326, 1998.

[26] C. Mahfoudi K. Djouani S. Rechak and M.Bouaziz. Optimal force distribution for the legs of an hexapod robot. *IEEE Conference on Control Application CCA Istambul*, 1:651670, 2003.

[27] C. Mahfoudi K. Djouani M.Bouaziz and S. Rechak. General method for dynamic modeling and control of an hexapod robot including optimal force distribution. *WSEAS, Transaction on Signal Processing*, 2:323330, 2006.

[28] W. Khalil and E. Dombre. *Modélisation identification et commande des robots*. Hermes sciences, Paris, 1999.

[29] T.watanabe and T.yochikawa. Grasping optimization using a required external force set. *IEEE transactions on automation science and engineering*, 4:52–66, 2007.

[30] S.Daniel ; EDIN Benoni B. Prediction of object contact during grasping. *Experimental brain research*, 3:265–277, 2008.

[31] X.Zhu and H.Ding. Computation of force-closure grasps. *IEEE Trans. Robot. Autom.*, 22(1):172–179, (2006).

[32] J.Gerardo and S.Mendez. *Sensors, Focus on Tactile, Force and Stress Sensors*. I-Tech, Vienna, Austria, Austria, December 2008.

[33] F. T. Cheng and D. E. Orin. Efficient algorithm for optimal force distribution-the compact-dual lp method. *IEEE Trans. on Robotics and Automation*, 6:178–187, 1990.

[34] Xiaoping Yun Eric Paljug and Vijay Kumar. Control of rolling contacts in multi-arm manipulation. *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION.*, (1994).

[35] Nilanjan Sarkar anf al. Dynamic control of 3d roling contacts in two arm manipulation. *" IEEE Trans. on Robotics and Automation*, page 364376., 1997.

[36] Q. Xiding and G. Yimin. Analysis of the dynamics of a six-legged vehicle. *" The International Journal of Robotics Research*, 14:1–8, 1995.